Masters Program in **Geospatial Technologies**

SENSOR FUSION OF IMU AND BLE USING A WELL-CONDITION TRIANGLE APPROACH FOR BLE POSITIONING

Amrit Karmacharya

Dissertation submitted in partial fulfilment of the requirements for the Degree of *Master of Science in Geospatial Technologies*

NOVA IMS Information Management School
Universitat Jaume I
Westfälische Wilhelms-Universität Münster

SENSOR FUSION OF IMU AND BLE USING A WELL-CONDITION TRIANGLE APPROACH FOR BLE POSITIONING

Dissertation supervised by:

Joaquin Torres-SOSPEDRA Ph.D.

Institute of New Imaging Technologies

Universitat Jaume I

Castellón de la Plana – Spain

Co-supervised by:

Prof. Dr. Cristian Kray

Institute for Geoinformatics

Westfälische Wilhelms-Universität Münster

Münster - Germany

Co-supervised by:

Dr. Mauro Castelli

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

Lisbon – Portugal

February 21, 2020

Acknowledgements

First and foremost, I would like to express my deep gratitude towards the **Erasmus Mundus Program** for the opportunity that I was bestowed upon with this masters degree. This program means to me a lot and my happiness knew no bound when I was awarded with this scholarship. Equally, I am very much obliged with **Ms Elena Martinez Santamaria** for providing me administrative support in each and every steps of my application and admission process. I can never remain aloof from being thankful towards **Dr. Joaquín Torres-Sospedra**, supervisor, **Prof. Dr. Christian Kray**, co-supervisor and **Dr. Mauro Castelli**, co-supervisor for their continuous support and guidance in each and obstacles I faced while achieving the objective of this project. Honestly speaking, without their incessant belief in me and motivation, this project would have never been completed. I am equally thankful with my professors for making me capable and worthy enough to complete projects like these. I would like to take this opportunity to be appreciative towards **Mr. German Mendoza Silva** for providing me with explanations and support. Your suggestions and experiences were highly valuable throughout the thesis. Additionally, I also like to thank the **staff of library and espaitec** who were involved in supporting me with the accessories required for testing and creating working environment. **Mr. Janak Parajuli**, is equally admirable for his efforts who was there for me to always support me whenever needed. **Dr. Tri Dev Acharya** for encouraging to work in a new topic. In this moment of cherish, I would also like to remember my father **Mr. Hira Prasad Karmacharya** and mother **Ms. Hari Maya Karmacharya** and family members in this auspicious moment. They have always done a lot for me and it is the result of their eternal sacrifice that what I am now. Finally, I would also express my deep gratefulness towards my **classmates, friends and relatives** who were always there for me in my thicks and thins. Without your unceasing love and generosity, I doubt this would have ever been possible. I am very much lucky to have you all in my life. A huge thank you once again.
...

Abstract

GPS has been a de-facto standard for outdoor positioning. For indoor positioning different systems exist. But there is no general solution to fit all situations. A popular choice among service provider is BLE-based IPS. BLE-has low cost, low power consumption, and tit is are compatible with newer smartphones. These factors make it suitable for mass market applications with an estimated market of 10 billion USD by 2020. Although, BLE-based IPS have advantages over its counterparts, it has not solved the position accuracy problem yet. More research is needed to meet the position accuracy required for indoor LBS. In this thesis, two ways for accuracy improvement were tested i) a new algorithm for BLE-based IPS was proposed and ii) fusion of BLE position estimates with IMU position estimates was implemented. The first way exploits a concept from control survey called well-conditioned triangle. Theoretically, a well-conditioned triangle is an equilateral triangle but for in practice, triangles whose angles are greater than 30° and less than 120° are considered well-conditioned. Triangles which do not satisfy well-condition are ill-conditioned. An estimated position has the least error if the geometry from which it is estimated satisfy well-condition. Ill-conditioned triangle should not be used for position estimation. The proposed algorithm checked for well-condition among the closest detected beacons and output estimates only when the beacons geometry satisfied well-condition. The proposed algorithm was compared with weighted centroid (WC) algorithm. Proposed algorithm did not improve on the accuracy but the variance in error was highly reduced. The second way tested was fusion of BLE and IMU using Kálmán filter. Fusion generally gives better results but a noteworthy result from fusion was that the position estimates during turns were accurate. When used separately, both BLE and IMU estimates showed errors in turns. Fusion with IMU improved the accuracy. More research is required to improve accuracy of BLE-based IPS. Reproducibility self-assessment (https://osf.io/j97zp/): 2, 2, 2, 1, 2 (input data, prepossessing, methods, computational environment, results).

KEYWORDS

Sensor Fusion
Kalman Filter
BLE positioning
Indoor Positioning
Smartphone

List of Abbreviations

ACCE	**A**ccelerometer
AoA	**A**ngle **o**f **A**rrival
AP	**A**ccess **P**oint
BLE	**B**luetooth **L**ow **E**nergy
ETRI	**E**lectronics and **T**elecommunications **R**esearch **I**nstitute
EvAAL	**E**valuating **A**mbient **A**ssistant **L**iving
GNSS	**G**lobal **N**avigation **S**atellite **S**ystem
GPS	**G**lobal **P**ositioning **S**ystem
GYRO	**GYRO**scope
IMU	**I**nertial **M**easurement **U**nit
INS	**I**nertial **N**avigation **S**ystem
IPS	**I**nertial **P**ositioning **S**ystem
IPIN	**I**ndoor **P**ositioning and **I**ndoor **N**avigation
k-NN	**k** **N**earest **Neighbor**
LBS	**L**ocation **B**ased **S**ervices
MAC	**M**edia **A**ccess **C**ontrol
PDR	**P**edestrian **D**istance **Reckoning**
RFID	**R**adio **F**requency **ID**entification
RSS	**R**eceived **S**ignal **S**trength
ToA	**T**ime **o**f **A**rrival
TDoA	**T**ime **D**ifference **o**f **A**rrival
USD	**U**nited **S**tates **D**ollar
UWB	**U**ltra **W**ide **B**and
WC	**W**eighted **C**entroid
Wi-Fi	**W**ireless **F**idility
WSN	**W**ireless **S**ensor **N**etwork
ZUPT	**Z**ero Velocity **UP**da**T**e

Contents

List of Figures

List of Tables

Chapter 1

Introduction

People spend about 80 percent of their time indoors [30]. In large offices and houses, people often feel the need for accurate positioning of the entities like employees chambers, restrooms, archive rooms etc. Even if people do have adequate knowledge of his/er surrounding, the provision of accurate positioning is always beneficial to have at the time of emergencies. Additionally in places like airports, libraries, museums, malls, and warehouses it is not possible to know every corner, especially for visitors and newcomers. The available positioning devices in present scenario i.e GPS/GNSS are primarily used in outdoor settings. Unfavorably in indoor environments, the accuracy of 4-5m provided by GPS is not enough as indoor activities require accuracy within the extent of a meter. GPS is not enough and even misleading in some cases as highlighted by [30]. Due to advancement in technologies several other systems have emerged which can provide position in buildings, or places where GNSS signals are weak or unavailable.

Indoor positioning is the process of positioning in indoor environments, where GNSS signals are not strong enough for positioning [14]. Indoor Positioning Systems (IPS) are systems devoted to provide indoor positioning. For example, Robotic vacuum cleaners use an IPS system to navigate around in a room. Novel uses of indoor positioning have appeared in health care where proximity interaction between individuals was studied to track the spread of influenza [39]. More indoor positioning applications are being developed [41, 42] and the indoor positioning market is predicted to be 10 billion USD by 2020 [9]. IPS can be highly accurate. For example, the Cricket's IPS could get centimetre level accuracy [36]. IPS utilizing Light, Sound, Ultra Wide Band (UWB), Artificial Magnetic Fields and Computer vision technologies achieve high accuracy [30]. However, accuracy is not the only determinant factor to assess an IPS system. Cost and scalability are significant factors too [10, 30] IPS that utilize already available Wi-Fi networks or cellular networks are cost effective and scalable, though they have lower accuracy than the technologies like Light or UWB. Researchers suggest smartphone based low cost, scalable solution with reasonable accuracy as a practical solution and demand for such a solution is high. Smartphones are seen as the best platform for mass market indoor positioning due to their ubiquity and convenience [44].

The past decade witnessed major developments in smartphone technology. Smartphones came with wide array of sensors. These sensors can be leveraged for position estimation. An example is the Mozilla's location service, which is “an open service which lets devices determine their location based on network infrastructures like Wi-Fi access points, cell towers and Bluetooth beacons." [32]. The Skyhook company provides "Precision Location" service using Wi-Fi, GNSS and Cellular network signals [38]. The composition of sensors in a smartphones vary by manufacturer and model. The precision of sensors also vary according to device. Nonetheless, smartphone sensors example Wi-Fi, Bluetooth, accelerometer, gyroscope, magnetometer, light sensor, proximity sensor have been researched for positioning [14]. Wang et al. [44] identifies dead reckoning, fingerprinting, trilateration, proximity estimation and visual localization as applicable methods for IPS in a smartphone. One or more of these methods can be used in combination as they are independent as well as complementary to each other.

IPS solutions based on only one technology are more prone to inaccuracies due to the sensor noise than those based on two or more technologies [45, 11]. Combining results from two or more technologies deliver better results than single ones [45, 48, 18, 23, 7]. It is common to see systems that combine network-based systems with inertial-based ones. Network-based IPS utilize Wi-Fi, Bluetooth, and Geo-magnetic field to provide position estimates. For all of them, new estimates are independent of the previous ones, which means that the error in one estimate does not affect subsequent estimates. Unlike Network-based IPS, Inertial systems utilize inertial measurements (acceleration and gyroscope readings) and are based on principle of dead reckoning. This means that the new estimates are based on previous ones. The error in a prior estimate is transferred to next estimate. This is known as drift error. Inertial systems can provide fast results continuously and consistently which is a desirable characteristic for real time systems. However, on long run, drift error accumulates and degrades accuracy of inertial systems. In network-based IPS, on long run, the system gets more data and positioning error is reduced. A hybrid system is able to utilize this contrasting feature to maximize accuracy of a combined system.

Many research proposals combine Wi-Fi or BLE with PDR to improve accuracy [48, 20, 37, 23]. Fingerprinting and trilateration are popular methods among network based systems. Wi-Fi is the most common IPS because Wi-Fi networks are easily available in most indoor environments and Wi-Fi fingerprinting provides good accuracy. But in practice, Wi-Fi networks are configured mainly to facilitate communication and not for positioning. Torres-Sospedra et al. [40] identified common alteration in AP infrastructure and showed that these changes have considerable impact on accuracy. Wi-Fi scan rate limitations imposed by newer smartphones also discourages Wi-Fi IPS. On the other hand, BLE is suitable alternative to Wi-Fi. BLE works similar to Wi-Fi and it is more accurate [15] with 1 to 2m accuracy [31]. BLE beacons need to be deployed in the service area. They are cheap, small, configurable devices with low power consumption that can last for months. Position of BLE beacons are known in advance and can be configured for positioning. BLE beacons may be deployed in high density network which helps in increasing accuracy. Due to these advantages [15], BLE is popular among IPS service providers and used for proximity application or relatively cheap positioning. Accuracy of the combined system is increased when accuracy of the individual system is increased. Proposed solutions or accuracy improvement for network based systems include creating high density fingerprint database, applying signal propagation model or using regression to enrich radio map [31].

1.1 Research Gap

For improving accuracy of BLE systems researchers have investigated diverse techniques. Linearizing non-linear beacon readings [23], applying stigmergy [34], channel diversity, Kálmán filtering and weighted triangulation [6] have been explored. Still, huge potential exists for further research because existing solutions are not flawless. For range-based BLE positioning, removing distant beacons from computation helps with accuracy [20, 30]. In range-based positioning methods, choice of beacons used for position estimation has direct influence on accuracy. It is known that RSS values suffer from some errors during transmission. This error in RSS causes error in computed distance. Weaker signal from a nearer beacons causes the beacons to be interpreted as being farther. This may contribute to error in final estimate. On the other hand, positioning usually requires more than one beacon to work with. Since, the positions of beacons are known in advance, closeness of each beacon is to other beacons can be known in advance. In a group of beacons, it is possible to know which beacons are closer to each other and which ones are farther. Farther beacons can be eliminated from position estimation. Since this elimination process depends on the actual distance between the beacons and not the RSS distance, one potential source of error is removed. In a beacons group, many permutations of beacons are possible but it is possible to choose a beacons permutation which provides the least possible error in the position estimate. According to a surveying concept, position estimation based on a well-conditioned triangle ensures least error[4]. To the author's knowledge this concept has not been explored so far in IPS. So, one of the goals of this thesis is to implement the concept of well-conditioned triangle for choosing beacons used in BLE position estimation.

1.2 Research Objectives

The research primarily aims to improve the accuracy of BLE-based positioning system.

To fulfil this objective, the specific objectives are

- To study and analyse factors affecting BLE positioning
- To design and implement a positioning algorithm based on well-conditioned triangle for BLE positioning and and study its effects.
- To implement an integration of BLE positioning and inertial method.

1.3 General Methodology

The thesis is divided into four stages namely; i) review and learning; ii) positioning using individual technologies, and analysis of new algorithm for BLE-based positioning; iii) sensor fusion; iv) comparison with existing methods.

In the first stage, several reviews of the IPS systems were studied. The focus of study was to gain overview of the current landscape of IPS. Given a broad field and numerous researches, Reviews of smartphone based and inertial based systems were prioritized. Existing BLE based and Inertial-based IPS systems along with combined systems were reviewed with focus on accuracy and implementation. Existing datasets and code were studied and reproduced to understand the working of different algorithms. Drawbacks of existing methods were identified and solution was conceptualized. The findings are discussed in chapter 2. Chapter 3 explains terminologies from control surveying which is a broad field related with computing precise horizontal and vertical coordinates

Second stage involved implementation of WiFi, BLE and IMU positioning estimates. A new algorithm for BLE positioning was also tested. A test track was designed to resemble a moving person. Tracking data were collected using GetSensorData mobile application. For Wi-Fi positioning, fingerprinting method was used. Training database for Wi-Fi fingerprinting were collected using another application. Test data was generated from the data collected during tracking. For BLE positioning, four factors that affect position estimate were studied. BLE observations were grouped by time windows. Size of the window and grouping technique would affect the position estimates. Different window sizes and different grouping techniques were studied. Moreover, each window may contain redundant advertisements. Strategies to handle redundant advertisements were also studied. A new algorithm for BLE positioning was designed and implemented. For IMU positioning, Stride-length and heading method was used. Stride-lengths were computed from accelerometer data and headings from gyroscope data. IMU estimates were sensitive to sensor bias and step detection threshold. Several test were conducted for fine-tuning. Explanation of the test environment, collected datasets and the mentioned methods are included in chapter 4. Chapter 5 elaborates the results and experiences and provides discussion on the obtained results.

Third stage is the sensor fusion stage. Kálmán filter was used to combine the BLE position estimates with IMU estimates. Kalman filter works on alternating sequence of predict and update phases. IMU position estimates were used for prediction phase and BLE position estimates were used for update phase. Since, IMU outputs are faster than BLE. Many IMU outputs occur between any two BLE outputs. This sequence was handled by creating a event-trigger list. This is a sorted list on timestamps of position estimate outputs. It diverts the Kalman Filter to predict or update phase depending on the source of position estimate. Chapter 4 contains brief explanation of the method and chapter 5 presents the result from fusion.

Fourth and final step is comparing the results with existing methods. Average error and 75th percentilele are computed as some literature report average error while the EvAAL-ETRI framework uses 3rd quartile. For uniformity, all combination of parameters are designed to output estimates at pre-configured time intervals. Comparative analysis and findings are discussed in chapter 5.

Chapter 2

Literature Review

This chapter presents a brief overview of existing indoor positioning systems in the context of network-based and inertial-based systems. The first section deals with the researches on indoor positioning techniques along-with their pros and cons. Second section discusses network-based positioning system. Reviews are focused on BLE-based IPS. Third section contains reviews of Inertial-based positioning system.

2.1 Indoor Positioning System

Large efforts have been dedicated to find new solutions for indoor positioning in the last decade. Meta-review [30] of IPS provides comprehensive insight into current IPS works and positioning techniques. More than 3900 unique works with doi numbers were found, but the real numbers is much larger as the review did not include papers without doi numbers. Light, computer vision, sound, magnetic fields, UWB, Wi-Fi, BLE, RFID, IMU, temperature, proximity [44] have been explored for positioning. High accuracy were achieved with Light, Sound, UWB, Artificial Magnetic Fields and Computer vision technologies [30]. Various surveys shed light on state of art techniques in specific domains. Correa et al. [10] focus on mass market applications, Dhobale et al. [12] reviews from user prospect, Diaz, Ahmed, and Kaiser [13] narrows down on inertial sensors, Basiri et al. [3] study usability and requirements. Wi-Fi was the most prevalent technology followed by light and Wi-Fi is predicted to remain dominant unless cheaper solution are found [30, 10]. But newer smartphones (android 9.0 and ios) have restricted Wi-Fi scan. This might cause decline in Wi-Fi IPS [30, 15]. Smartphones are seen as the perfect device for mass market positioning systems. Low cost and smartphone based solution have high demand[48]. Light and BLE based IPS are currently runner-up in terms of research, but BLE is popular among IPS providers [30]. Accuracy is not the only important factor for choosing an IPS. Infrastructure cost and scalability to all environments are significant too [10]. Coverage, Complexity, robustness, privacy and power consumption are also considered [30]. Due to many parameters, there is no clear winner, many solutions have been proposed adhering to different environment and applications.

Dead reckoning, fingerprinting, trilateration, triangulation, proximity estimation, visual localization are popular techniques for indoor positioning. Combination of one or more of these techniques is also possible. Such systems are called hybrid systems or fusion systems. LearnLoc [35], Kailos [17], Surround Sense [2] are some hybrid methods. A comprehensive discussion on hybrid systems is provided by Easson [14]. Each of these techniques have advantages and disadvantages. Dead reckoning is accurate but drift error accumulation is a major challenge [47, 13]. Fingerprinting is a widely used method but radio sensor error and maintenance of fingerprint database makes it labour-intensive and cumbersome [11]. Battery capacity of a smartphone has been a major consideration in development of smartphone based IPS [14]. Many systems perform computations in remote server instead of the smartphones [22]. Visual recognition based approaches suffer most as camera takes up considerable resource. This presents challenges for a resource constraint smartphone but its applications can be seen in robotics example is robotic vacuum cleaners. Visual odometers perform better than wheel odometers or low cost inertial sensors [1] hence they are suitable for robots. Proximity estimation are good for rough estimates but fail when accurate positioning is required. Limited range of proximity sensors reduce their usability [14]. In real world, IPS systems find uses in positioning systems, tracking systems and/or navigation systems.Positioning means to find a point with reference to a known reference frame. Example usage of positioning is to find the location of a product in a warehouse. Tracking systems involve positioning over time. It requires a dynamic process model and measurements of the tracked object [45]. Tracking is effective for objects which can make turns and for situations when positioning estimates are uncertain. Navigation system on the other hand suggest a path from an origin to destination. Suggestions are usually based in underlying data, navigation algorithm and system configuration. Navigation system actively suggests a path whereas, tracking system records previous path. Potential uses of these systems contribute to rise in demand and research for IPS. Research landscape is focused on experimentation with different combinations of technologies. Target of research is seen on increasing accuracy and lowering cost.

2.2 Network-based systems

Wi-Fi, BLE, UWB, Cellular network, WSN, Geo-magnetic field are examples of network-based systems. Only Wi-Fi and BLE based systems ave been reviewed for this work.

Network-based method can further be classified into range based method and range free methods. Range based methods, example trilateration and triangulation, determine angle or distance from network signals and use geometric computations to find position [10]. Range free methods, example fingerprinting and proximity estimation, exploit patterns that are position dependent. Fingerprinting is a widely used range-free positioning technique and provides accurate results [30]. A fingerprint is the collection of RSS values from different APs at any point. This fingerprint is different at different points. It is characteristic of the place where it was collected and it can be exploited to determine position. In fingerprinting, RSS fingerprint observed at some point is compared with a database of previously collected fingerprints to find a match or closest match/es and using them to determine position. It is performed in two steps: firstly some selected positions with known coordinates are marked with unique ids, fingerprints at those positions is captured and stored in a database. In the second step the positions is unknown but the device can capture RSS and generate a fingerprint. This fingerprint is compared with the database collected in previous step to find the closest match/es. Denser the database, higher the position accuracy. Creating, maintaining and updating dense database has been challenging part for fingerprinting applications.

Trilateration is a range-based positioning technique that can work with fewer number of APs than fingerprinting. It involves more computation but it can work with three APs. Depending on method of distance estimation trilateration can further be divided as RSS distance estimation, Time of Arrival (ToA) or Time Difference of Arrival (TDoA). If the transmitted power is known, RSS value can be used to estimate distance [26]. Signal attenuation models facilitate distance estimation but signal propagation in indoor environment is not ideal. However, empirical model which use RSS value as log-normally distributed random variable are supported by empirical evidence [26]. Using empirical model, a maximum likelihood estimation of distance for a given RSS measurement can be computed. ToA and TDoA require smartphone to send signal to AP and estimate distances using round trip time. They are more accurate than fingerprinting but are more affected by time delays. Newer research employ Angle of Arrival (AoA) method. Ye et al. [48] proposed a single AP based system using multiple antennas. It used Angle of Arrival (AoA) and gives high accuracy at low cost. Trilateration techniques although more accurate are not popular because of complexity in angular measurements, distance and time delays [15]. Weighted Centroid (WC) is another technique for positioning. WC requires prior knowledge on positions of the signal sources and a way to identify them from RSS signals. WC computes weighted average of positions of the detected sources [33]. The algorithm is explained in the datasets and methods chapter. The BLE research works found weighted centroid better but more sensitive to disconnections than fingerprinting [31, 25].

Wi-Fi is in the forefront of indoor positioning but newer changes make further progress challenging. Faragher and Harle [15] identify five challenges:

1. increasing passive scan duration,
2. buffered aggregate reporting,
3. increase in network traffic due to active scans,
4. non-compatibility in all mobile platforms
5. lack of standard unit for reporting signal strength

Newer smartphones (android 9.0 and ios) have restricted Wi-Fi scans to 4 scans every 2 minutes which is too low for positioning [30]. Wi-Fi sensor buffers received signals and outputs a single aggregate report. If user is moving during buffer period, the aggregate is smeared with multiple positions. Long scan duration further increase the smearing effect. Active scans engage APs and increase network traffic, hence the total throughput of network is reduced. Apple devices do not allow RSS reading of APs except the one to which it is connected. It also cannot connect to more than one AP at a time. Hence, fingerprinting in Apple devices is not possible. Wi-Fi specification does not specify a unit for reporting signal strength value. Systems configured in one unit may not perform or perform in unexpected manners with devices using other unit. No survey covering BLE based IPS are available [30] but it is seen as the suitable positioning technology for smartphones [10]. Identified advantages of using BLE are:

1. Advertisements are reported immediately
2. Standard unit for reporting is specified as dBm [5]
3. Lower power consumption almost half of Wi-Fi [24]
4. Low cost per beacon
5. Short setup time compared to fingerprint collection

Selection of signal source is known to have influence on the accuracy of the system. In EZ system [8], found selecting correct AP, selecting subset of location and dividing wall problem as major issues in the system. LocSelect [8] showed selecting subset of location based on some design example RSS information overlap provided better result. Kang et al. [20] proposed a hybrid method where distant beacons are filtered out to improve accuracy. Mendoza-Silva, Torres-Sospedra, and Huerta [30] also agrees on deteriorated accuracy when distant emitters are used.

2.3 Inertial based systems

Accelerometer, magnetometer gyroscope comprises the inertial sensors. The sensors are tri-axial i.e. they report in x,y and z axis. Note that the axes are relative to the device and not to the earth axes. A data that combines accelerometer and gyroscope is said to have 6 degrees of freedom and addition of magnetometer is said of have 9 degrees of freedom. The accelerometer is used to compute distances and velocity of the device and gyroscope and magnetometer are used find orientation, heading and rotation of device. Once initial position is known, an inertial system can track without external influences. There is high interest in this method as these sensors are readily available in smartphones. The process most used by inertial sensors is Pedestrian Dead Reckoning (PDR). Easson [14] recognize it as excellent strategy. PDR has shown sufficient accuracy but its major drawback is the drift error. Most errors occur due to biases, bias stability and thermo mechanical noise [47]. Woodman [47] define bias as measured average value of the acceleration and turn rate when neither acceleration not rotation is undergoing. It can be compensated by averaging values from sensors in static mode and subtracting averaged value from measured values. This is also known as calibration. Bias stability is how biases change over time under stable conditions at constant temperature. This causes non-systematic error which is difficult to compensate as biases change over time. Thermo-mechanical noise introduces random noise which has no correlation to the sensors.

In PDR displacement and direction from a known position is used to compute new position. A position can be estimated using two approaches: Strap-down method and Stride-length and heading method. Strap-down methods filter the incoming accelerometer and magnetometer data and use double integration of acceleration to compute positions. An example of strap-down method is Zero-velocity UPdaTe(ZUPT) algorithm. It is common in shoe-mounted IPS. Shoe mounted systems can accurately detect step and re-calibrate in each step [13]. This in turn helps to limit drift error. In inertial systems drift error accumulation is cubic with time [21]. ZUPT algorithm was able to convert the cubic error growth to linear [13]. This reduced the drift error in ZUPT systems but the linear error accumulation was still found to be challenging. Despite lower error, shoe mounted systems require additional sensors to be mounted in shoes. Due to this their practical applications are somewhat restricted. Stride-length and heading method employ some key stages: i)

Step detection, ii) Stride-length estimation, iii) Heading estimation and iv) Position Computation. Computed stride-length and heading are applied on previous position to get new position estimate. There are many variations of stride-length method. Vezočnik and Juric [43] evaluated 13 different models of stride-length and heading method. Models were evaluated on different walking speed (slow , normal and fast), position of device (Pocket, bag, hand-reading and hand-swinging) and model parameters (personalized and universal). Most modals performed better for fast walking speed for personalized parameters. Weinberg model performed overall best when personalized set of parameters were used. IPS solution that can function with Smartphone but without external sensors are of high interest. ZUPT systems are not practical for smartphone only IPS. ZUPT require detecting zero-velocity for which it needs an external foot-mounted sensor. Instead, stride-length and heading algorithm are feasible solutions for smartphone only systems.

2.4 Sensor Fusion

Sensor fusion enables to control drift error in an IPS [48, 23, 45, 7, 49, 50]. IPS systems are complementary to each other. Wang et al.[45] found Tracking of moving target was better with fusion and recommends adding another sensor for better performance. Zihajehzadeh et al. [50] states that fused system can maintain tracking during GPS outrages for 5 second with error less than 2 m. Chen et al. Chen et al. [7] fused Wi-Fi and inertial sensors and improved them using landmarks. This shows that fusion is not limited to sensors. Combination is possible with many technologies. Map matching is a powerful technique if the layout of place is already known. Popular techniques for fusion are Kálmán Filter and Particle Filter.

Kálmán filter are based on Gaussian filtering. The system at any state is represented by state variables for example the position. The next state can be reached using two methods, a theoretical model whose uncertainties can be computed and another is observation model which can determine the state within certain error. Both of these models contain uncertainties, Kálmán filter is method of combining both of these uncertainties to generate a result with higher precision than each of them.

The first step is prediction step, where the filter estimates new position with and its uncertainty measures. Next step is update step, where observed state is taken and compared with the estimates from previous step. The observed state also has uncertainties. Weights for each model are updated based on uncertainty with higher certainty getting higher weight. At the start one of the model is taken as accurate (usually observation model) and assigned full weight. This weight is updated in each update step.

Particles filters are based on solution of Bayesian filtering [16]. A set of weighted random samples or particles is used to represent the initial condition. These particles are distributed over the building. Constraints like walls, forbidden places are used to filter out particles. Stairs and lifts can transition particles to other levels. Particles that collide with walls are excluded from the simulation. The simulation is recursively solved for final position. User location is weighted mean of remaining particles. It can be represented in four steps 1) Prediction 2) Application of constraints 3) Update and 4) Re-sampling. Particle filter are natural and intuitive ways of including map layouts and applying constraints directly on the particle. The map can be represented as Rooms, Stairs and Ladders / Elevators. Addition of constraints improve the results, but it is difficult to modal large open areas as number of particles required is more. Particle filter was popular in track 1 of IPIN conference 2018 [37].

Chapter 3

Theoretical background

This chapter introduces terminologies borrowed from the domain of surveying specifically control surveying. Control surveying deals with determination of precise position using distance and angular measurements. Trilateration, the principle on which GPS/GNSS is based one of the technique control survey. This work implements some concepts from surveying, so they are explained here.

3.1 Resection

Resection is a method of determining an unknown position using measurements to three known positions. Angles or distances or their combination can be measured. Example every point on a line segment can be determined if the distance to the end vertex are known or the ratio of distances from each vertex is known. Similarly, for a triangle any point within and on the plane of the triangle can be determined if the distance from each vertex or ratio of distances from each vertex are known. Similarly, the concept can be extended to higher geometries. In real life, true value of a distance or an angle can not be accurately measured. All measurements include some error. Even careful measurement using precise devices and observers still contains some error. This error inflicts more error during computation. Given this fact, in resection it is accepted that the accuracy of the unknown point is higher if the geometry from which it is derived is well-conditioned.

There seems to be confusion when using the term triangulation, trilateration and resection. In triangulation and trilateration, measurements are undertaken from known positions to unknown positions. In resection, measurements are carried out from unknown position to known positions. For distance measurements, distance from unknown to known and known to unknown are theoretically same so the term trilateration and resection can be used interchangeably as the computation technique will remain same. For angle measurement though, angle from known to unknown and from unknown to known is different in value. So use of the terms triangulation or resection is different. Example in beam-forming [28], angle is measured at antenna array. The position of antenna array is known but position of target is unknown. this corresponds to case of triangulation. If target position is unknown and the measurements are made from the unknown position, it corresponds to resection. Literature on IPS have not used the term resection so far to the author's knowledge. Some literature have used the term trilateration to mean position computation using distance to three known points and multi-lateration to mean computation using distance to more than three known points. For consistency with existing IPS terminologies, this document uses the term trilateration but it can be interpreted to mean resection where suitable.

3.2 Well-conditioned triangle

In trilateration error in position computation depends on the geometry formed by the three known points. The shape of triangle formed by those three points affect the accuracy [4].

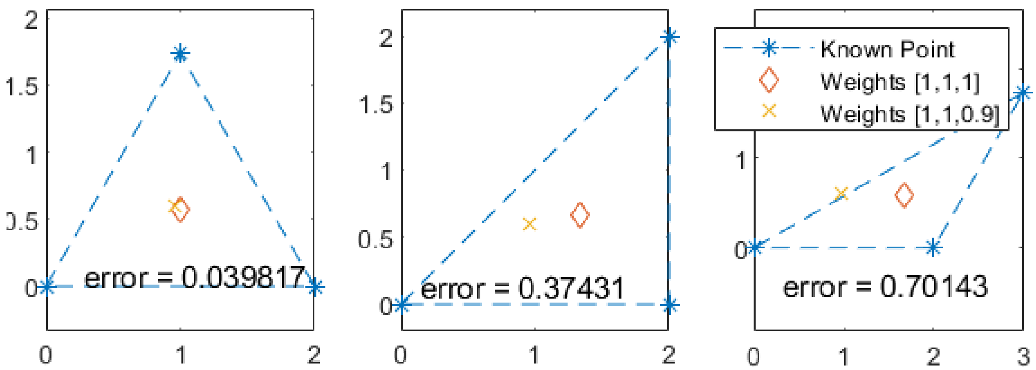


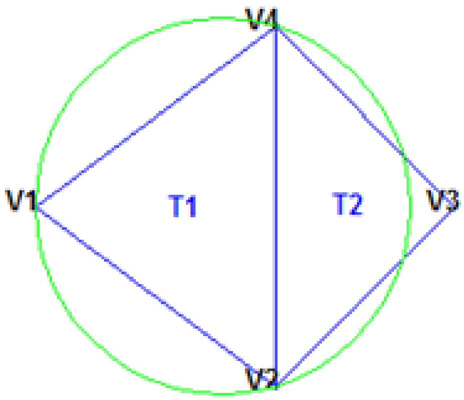
FIGURE 3.1: Error for same change in weights for different triangle geometry

In figure 3.1 Consider the known points (example BLE beacons) are represented by blue stars. Two of the sides are of unit 2 and length of the third side is dependent on geometry. Suppose that in ideal condition with no multipath and ideal path loss, the readings can be equally trusted so they are given weights of [1,1,1] respectively (in practice in second figure the vertex on bottom-right should have more weight as it is nearer, but we assign same weight for simpler computation). The unknown position is calculated as weighted average of the known points. The calculated position from the above weights is shown in diamond. Now suppose due to some error in signal from the upper point, it is trusted a bit less. Now the weights become [1,1,0.9]. Position computed with these weights are shown in cross. From the figure it is evident that the error in position due to change in weight is least for an equilateral triangle and the position error increases as the triangle deviates towards scalene.

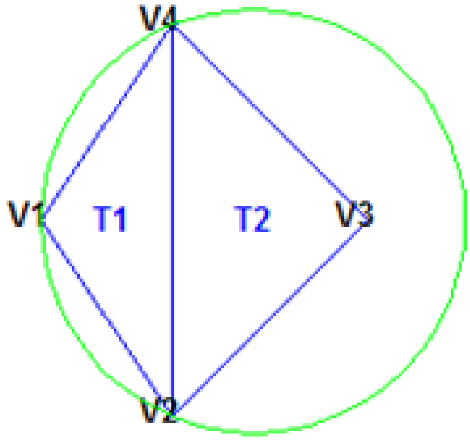
Theoretically in an isosceles triangle with two angles of 56° 14', change in any measurement (distance or angle) to unknown point will have least effect on the resulting position. Such a triangle is known as a well-conditioned triangle [4]. This value takes one side as base for computation. But a triangle can be solved from other sides as well so the best geometry is an equilateral triangle. The worst geometry is a straight line. A straight line formed by 3 points will make a triangle with angles of 0°, 0° and 180°. In practice equilateral triangles are rare, so a well conditioned triangle is defined as a triangle in which no angle is less than 30°. Triangles having angles less than 30° are considered to be ill-conditioned and should not be used for computations.

3.3 Delaunay condition

A Delaunay triangulation is a set of lines joining a set of points together such that each point is joined to its nearest neighbors. The set of lines form a triangular mesh. In this triangular mesh every circum-circle of a triangle does not contain any other points of the set within it. Delaunay condition states that the circum-circle of any triangle should not contain any other point inside it. Circum-circle is the circle that passes through the vertices of a triangle. A triangular mesh satisfying delaunay condition is called delaunay triangulation. For example, from a set of four points it is possible to form four different triangles. Among the four triangles only two triangles will satisfy Delaunay condition. It is a property of delaunay condition that the triangles are selected in such way that the minimum internal angles of the selected triangles are as large as possible. Due to this property, the member triangles are considered well shaped. This is important property as maximizing the minimum angle favors well-condition.



(A) Triangles satisfying Delaunay Condition [27]



(B) Triangles not satisfying Delaunay Condition [27]

FIGURE 3.2: Possible triangulation mesh with four points

In figure 3.2a, the triangles satisfy the Delaunay condition as the circle does not have any points in them. In contrast to previous one, the circum-circles of the triangles in figure 3.2b have points in them and hence these triangles do not satisfy the Delaunay condition. It can be observed that these triangles have sharper angles nodes V2 and V4 than the previous ones. There is always the possibility to convert this triangulation into Delaunay by replacing the edge V2-V4 with V1-V3 as this would increase the minimum internal angles and fulfillment of the Delaunay condition can be expected. Another important property of Delaunay condition is it uses nearest-neighbor relation to connect the points. This enhances the implications of Delaunay triangulation in data interpolation as well. The concept of Delaunay triangulation for 3D is also similar, only the circum-circle is replaced by circum-sphere and triangulation by tetrahedrons.

This chapter explained the new terminologies, it shed light on a naming confusion between IPS field and surveying field. The concept of well-conditioned triangle and it's effect on position error has been shown. Delaunay condition has been discussed for getting well-conditioned triangles.

Chapter 4

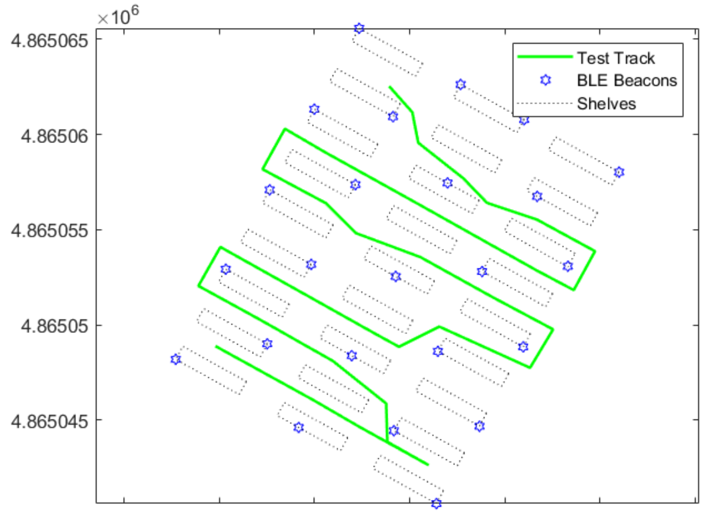
Methods and Data-sets

This chapter discusses the Data-sets and methods used. It describes the experiment area and setup, data collection process, collected datasets and how they are processed. Methods for determining position form Wi-Fi, BLE and IMU are presented. Proposed method for BLE is described. Sensor fusion approach used to combine the position estimates is presented in current context. Finally, method used to compute error is described.

4.1 Test Environment

The experiment area is a wing of a university library. Measurements are carried out in the 5th floor of the library where the BLE beacons had been deployed. This area is among bookshelves which can block RSS signals. The area of experiment is about 176 sq. m. and height of the floor ceiling is around 2.65 m and height of shelves is 2.35 m. Shelves cover 68.73 sq. m. The Wi-Fi APs are installed on the ceiling [29]. The beacons were placed in the top of the book shelves. They are not visible from outside. 22 BLE beacons were deployed in the area. The deployment resembles a dense distribution (1 beacon per 7.86 sq. m.). It had been designed with the goal of supporting a positioning service. The area covered by the beacons is 172.9 sq. m. and covers all the book shelves [31]. The area and device settings are same from [29] and [31].

BLE beacons used are Accent Systems' IBIK 105. They can broadcast iBeacon TM and Eddystone TM advertisements concurrently over different emission slots. In this work they have been configured to use one iBeacon slot with advertisement period of 200ms. Advertisements broadcast on three channels in quick succession.



(A) Plan of shelves, beacons and test track in Experiment Area

(B) Beacons placed on the top of book shelves [31]

4.2 Data Collection

Wi-Fi fingerprints training dataset were collected specifically for this work. The dataset was collected by the author using an existing fingerprint collecting software specific for the test environment. The collection occured at predetermined points, wherein the collector had to face a specific direction while holding the collection device in front of his chest. The software was setup with a 260 points grouped into 5 campaigns. Each campaign was an ordered list of points in forward and reverse direction. 6 fingerprints were collected at each point facing forward direction and 6 more were collected facing opposite direction. The software stores raw data in json format. Listing 1 shows a sample of the collected raw data. Raw measurements were uploaded to server which created final dataset. Wi-Fi fingerprints training data collection process is described in detail in the Long term Wi-Fi fingerprinting dataset paper [29]. The newly collected training dataset consisted of 180 unique Wi-Fi APs and 3120 fingerprints. Table 4.1 shows a snapshot of the database. The first row is the mac address of the detected APs. Mac address have been replaced by an identifier in the figure. Only four columns are presented in the figure due to display space constraint. Subsequent rows list the RSS value detected from corresponding AP and its position. APs not detected in fingerprint are given value of 100 [29].

The test data is captured with Samsung s6 smartphone (Model: SM-G920F) running on android version 7.0 and API Android version 24. The application used for data collection is *GetSensorData* version 2.1 which is also used by IPIN conference since 2016 [19].

```
[
  {
    "campaignPointDetails": { "id": "1" },
    "fingerprints": [
      {
        "androidVersion": "7.0",
        "dateCaptured": "20191031153734798",
        "dateStored": "20191031153734798",
        "listOfRSS": [
          {
            "channel": 5220,
            "intensity": -57,
            "mac": "06:18:d6:03:27:aa",
            "position": 0,
            "ssid": "UJI"
          },
          {
            "channel": 2412,
            "intensity": -61,
            "mac": "04:18:d6:04:27:aa",
            "position": 1,
            "ssid": "eduroam"
          }]
        :
        :
        :
        :
```


LISTING 1: Raw data captured during Wi-Fi fingerprint collection

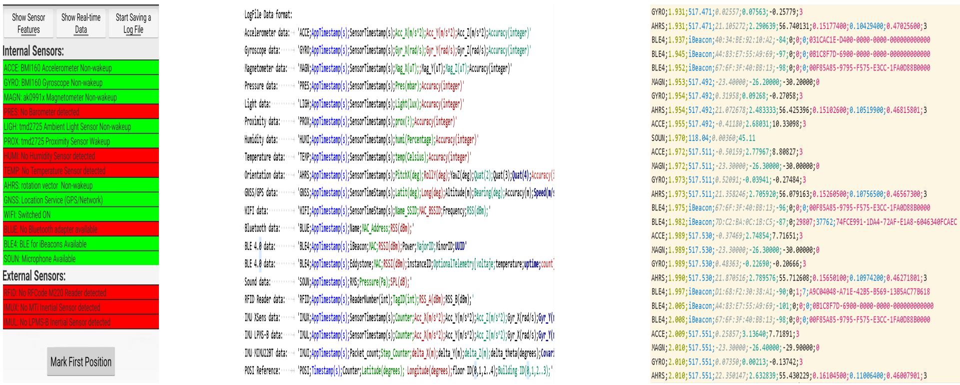
The application captures data from android smartphone sensors and outputs them in a log-file. It supports capturing internal sensors accelerometer, gyroscope, magnetometer atmospheric pressure, ambient light, proximity, humidity.etc as well as from attached external devices example RFID reader, XSsens IMU or LPMS-B IMU devices [19]. Since IMU measurements suffer from sensor bias, usually a calibration step is required where measurements are collected when the device is static and bias values are determined. GetSensorData app does not have a calibration option. But it has option to mark locations. A workaround devised was to stand still for some seconds before starting to walk.

Data collection proceeds in this way. At the start of the test track and the device is static. First, button to start saving a log-file is pressed but movement is not started immediately.

mac1	mac2	...	mac179	mac180	X	Y	floor	timestamp
-61	-92	...	100	100	-7746.9613	4865057.377	3	2.0191E+16
-63	-89	...	100	100	-7746.9613	4865057.377	3	2.0191E+16
-62	-89	...	100	100	-7746.9613	4865057.377	3	2.0191E+16
-61	100	...	100	100	-7746.9613	4865057.377	3	2.0191E+16
-61	100	...	100	100	-7746.9613	4865057.377	3	2.0191E+16
-61	-92	...	100	100	-7746.9613	4865057.377	3	2.0191E+16
-55	-87	...	100	100	-7751.6529	4865060.127	3	2.0191E+16
-55	-87	...	100	100	-7751.6529	4865060.127	3	2.0191E+16
-59	-83	...	100	100	-7751.6529	4865060.127	3	2.0191E+16
-59	-90	...	100	100	-7751.6529	4865060.127	3	2.0191E+16
-59	-88	...	100	100	-7751.6529	4865060.127	3	2.0191E+16
-59	-85	...	100	100	-7751.6529	4865060.127	3	2.0191E+16
-60	-81	...	100	100	-7756.5313	4865062.868	3	2.0191E+16
-59	-83	...	100	100	-7756.5313	4865062.868	3	2.0191E+16
-60	-82	...	100	100	-7756.5313	4865062.868	3	2.0191E+16
-59	-82	...	100	100	-7756.5313	4865062.868	3	2.0191E+16
-60	-82	...	100	100	-7756.5313	4865062.868	3	2.0191E+16
-61	-83	...	100	100	-7756.5313	4865062.868	3	2.0191E+16
-70	100	...	100	100	-7748.1303	4865055.303	3	2.0191E+16
-72	100	...	100	100	-7748.1303	4865055.303	3	2.0191E+16
-74	-91	...	100	100	-7748.1303	4865055.303	3	2.0191E+16
-73	-91	...	100	100	-7748.1303	4865055.303	3	2.0191E+16
-75	100	...	100	100	-7748.1303	4865055.303	3	2.0191E+16
-71	-91	...	100	100	-7748.1303	4865055.303	3	2.0191E+16
-66	100	...	100	100	-7752.7494	4865058.019	3	2.0191E+16

TABLE 4.1: Snapshot of Wi-Fi Fingerprint database

Instead, the device is kept static for few seconds. This is to collect calibration data for IMU. After few seconds, button to mark position is pressed and movement is started. In the log-file first mark would signal start of movement. Measurements before first mark are considered static and used for determining sensor bias. Afterwards, while moving positions are marked at pre-determined points. At end of track, button to stop saving log-file is pressed. An experiment was set up to simulate a moving user. A test track was used as a reference in the library. The track is shown in figure 4.1a in green. It would describe the path of a person moving in the library between bookshelves. The collecting device was held in hand in front of the chest all time during data collection. The movement starts from the upper end of the track in Figure 4.1a).



(A) Application Interface (B) Data Format (C) Sample data

FIGURE 4.2: GetSensordata Application and log-file: c shows output from GetSensorData app and b shows, depending on the sensor, how each row in c should be interpreted

A format of the log-file is presented in figure 4.2b and a sample of the data captured in figure 4.2c. The log-file stores data from particular sensor and in a particular time in a separate row in a sequential order. First four character of each row is a code for sensor example ACCE for accelerometer, AHRS for rotation vector, BLE4 for Bluetooth low energy. It should be noted that BLUE is for classic Bluetooth while BLE4 is for Bluetooth Low Energy. Classic Bluetooth is not captured in this experiment. After the four characters, remaining line is the data. Each value is separated by a semi-colon. The values are in the order described in the format. For example, first value of first row 'GYRO' means the subsequent values in first row are AppTimeStamp (1.931), SensorTimeStamp (517.471), Gyr_x (0.02557), Gyr_y (0.07563), Gyr_z (-0.25779) and accuracy (3). The units are specified in the format. A Matlab parser is also provided by the application developers to process the log-file and separate entries by source sensors.

4.3 Methods

4.3.1 Wi-Fi positioning

K Nearest Neighbors (k-NN) fingerprinting algorithm was applied for position estimation. New collected fingerprint database was used for position estimation. Collection process of this database has been described in data collection section. During the experiment, sensor data are captured by *GetSensorData* app in a log-file. The log-file was transferred to a computer where data processing and analysis was done. Wi-Fi data are separated out from the log-file in matlab using the provided data parser. The data looks as in figure 4.3a. It consisted of AppTimeStamp, SensorTimeStamp, mac, frequency and RSS values for each row. For position estimation a fingerprint has to be generated from this data. Here, a fingerprint is an ordered collection of RSS values arranged in the same order of mac address as the database. This is because the order refers to the value captured for a specific AP. The RSS values should be from same timestamp. The order of mac address was retrieved from the database. Wi-Fi sensor outputs aggregate report so, the values output in a report has same AppTimeStamp. This AppTimeStamp was used to group RSS values into a fingerprint. All rows having same AppTimeStamp are kept in one group. AppTimeStamp is used instead of SensorTimeStamp as SensorTimeStamp shows fluctuations. The group is then ordered into a vector matching the order of the database. Corresponding fingerprint of data in figure 4.3a is shown in figure 4.3c.

AppTimestamp	Sensortimestamp	MAC	Frequency	RSS
0.027000	515.52	6.7037e+12	5180	-52
0.027000	515.52	4.5047e+12	5180	-85
0.027000	511.68	6.7037e+12	5220	-90
0.027000	515.52	6.7037e+12	5180	-86
0.027000	515.52	6.7037e+12	2412	-87
0.027000	515.52	4.5047e+12	2412	-87
0.027000	515.52	6.7037e+12	2462	-59
0.027000	515.52	4.5047e+12	2462	-59
0.027000	515.52	3.0709e+12	5600	-91
0.027000	511.68	4.5047e+12	5180	-84
0.027000	515.52	6.7037e+12	5180	-88
0.027000	511.68	6.7037e+12	2412	-89
0.027000	515.52	4.5047e+12	2412	-87
0.027000	515.52	4.2487e+13	2462	-89
0.027000	515.52	4.5047e+12	5180	-56
0.027000	515.52	2.2880e+14	2467	-84
0.027000	515.52	4.0288e+13	2462	-89
0.027000	515.52	1.1438e+11	2452	-89
0.027000	515.52	1.6794e+11	2472	-87
0.027000	515.52	1.1438e+11	2452	-88

(A) Wi-Fi RSS group

	1
1	6.7037e+12
2	4.5047e+12
3	6.7037e+12
4	6.7037e+12
5	6.7037e+12
6	6.7037e+12
7	4.5047e+12
8	4.5047e+12
9	4.0288e+13
10	4.0288e+13
11	4.5047e+12
12	6.7037e+12
13	4.2487e+13
14	6.7037e+12
15	4.2487e+13
16	6.7037e+12
17	1.1263e+14
18	6.7037e+12

(B) Database Mac Order

1	-88
2	-87
3	-110
4	-89
5	-52
6	-59
7	-110
8	-110
9	-110
10	-89
11	-110
12	-90
13	-89
14	-110
15	-110
16	-110
17	-110
18	-110

(C) Wi-Fi Fingerprint

FIGURE 4.3: Wi-Fi Fingerprint generation from data. a RSS data collected at certain time, b Order of MAC in fingerprint database, c Resultant fingerprint.

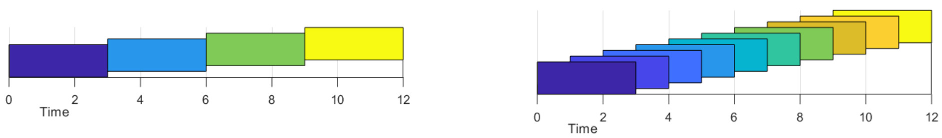
This fingerprint was matched with other fingerprints from the database collected in previous step. K Nearest Neighbours (k-NN) algorithm was applied for match finding. In k-NN method, k nearest fingerprints are selected from the database based on distance in fingerprint space. If the captured fingerprint exactly matches with any fingerprint then location of the match was used. Else the average of the positions of the selected fingerprints were used.

4.3.2 BLE positioning

Wi-Fi sensor aggregate RSS from all the APs but BLE sensor registers advertisements anytime an advertisement is detected. Wi-Fi advertisements could be grouped by same timestamp value but BLE advertisements do not have a common timestamp. Each BLE advertisement has its own timestamp value. Positioning requires analysing a group of advertisements that are closer in time. This made it necessary to group advertisements by time window. Two techniques of grouping window have been used namely i) Discrete Time Window and ii) Continuous Time Window. In a window, it is possible to detect advertisements from the same beacon more than once. Such redundant advertisements were handled to give only one value. Handling of redundant advertisements could be carried out by any of these three methods (a) last of advertisements, (b) average of advertisements or (c) highest of advertisements.

Discrete Time Window

Advertisements were collected in buckets for certain time. The time was set by the window size example 1 second, 2 seconds, 1.5 seconds. Advertisements detected within a time window were grouped in same bucket. Then the advertisements in the bucket were used to compute position. After the time for a bucket expires, new bucket was created and the process was repeated. It is possible that more than one advertisement from same beacon are observed in same window. In such case, those redundant advertisements need to be processed. This technique uses lower computing resources but the position in mid of a window is unknown.



(A) Discrete Window (B) Continuous Window

FIGURE 4.4: Bucket in Discrete and Continuous grouping technique

Continuous Time Window

In the discrete time window technique, advertisements at 0.1 second and 0.9 seconds group into same bucket whereas advertisements at 0.9 and at 1.1 would be separated into different buckets (if window size is 1s). This creates a problem as the two latter advertisement are closer in time than the former two. This problem was solved using continuous time window technique. Continuous window technique features an extra update interval parameter on top of discrete window technique. The update interval parameter dictates how often position is computed. In this technique, a bucket is created at set intervals e.g every 0.1 second or 0.5 second. Each bucket will collect advertisements for a certain time set by window size. Update interval is kept lower than window size so buckets overlap each other. An advertisement may fall in many buckets. When a bucket expires, position is computed from the advertisements collected in that bucket. In continuous time window technique the two latter advertisements get grouped into same window more times than the former two. This way simultaneous advertisements have more effect. This makes continuous window more sensitive than the discrete method. Disadvantage is that this technique runs more frequently and processes the same advertisement multiple times. It requires more computation resources than discrete technique so it may not be suitable for low end processors.

Weighted Centroid for BLE

Weighted centroid method [33] is applicable where beacon positions (x_i, y_i) are known beforehand. The position estimate is given by the equation 4.1, using weights calculate by equation 4.2.

$$(x, y) = \sum_{i=1}^{k} \omega_i (x_i, y_i) \tag{4.1}$$

$$\omega_i = \frac{\omega_i^{'}}{\sum_{j=1}^{k} \omega_j^{'}} \tag{4.2}$$

where:

$\omega_i^{'}$ = weighting factor

The weighing factor was computed using an existing empirical model for the test environment. The model exploits the reduction of signal strength during transmission. It converts an RSS value to a weight value. Higher signal strength get higher weight and lower signal strength get lower weights.

Proposed method for BLE

A new method was proposed to select beacons used in positioning. Due to the error in the RSS value, any calculations that use RSS values are prone to transfer the error. This method is based on the beacons layout. Known distance between beacons were more precise than distances computed using RSS values. Calculations based on precise distances should be more reliable than those based on less precise distances. The method works in following way. First, list of detected beacons in a window were arranged in descending order of RSS value. This was to arrange beacons in order of proximity. Signal strength decreases with increasing distance so higher RSS value correspond to closer source. After that, 3 nearest beacons were selected and checked for well-condition. If they satisfied well-condition then position was estimated as weighted average of the selected beacons. If the 3 selected beacons did not satisfy well-condition, next proximal beacon was added to selection. It is possible to form four different triangle using combination of four points (4C3 = 6). With five points number of combination increases to 10 (5C3 = 10). An efficient way is required to reduce computation. Delaunay triangulation creates a triangular network that maximizes the minimum angle of any triangles in the network and the triangles do not overlap each other. This method is suitable to get well-conditioned triangles and reduce number of combinations. When multiple well-conditioned triangles were detected, weighted average of beacons in well-conditioned triangles was used as estimate. If no well-conditioned triangle are detected, next closest beacon is added.

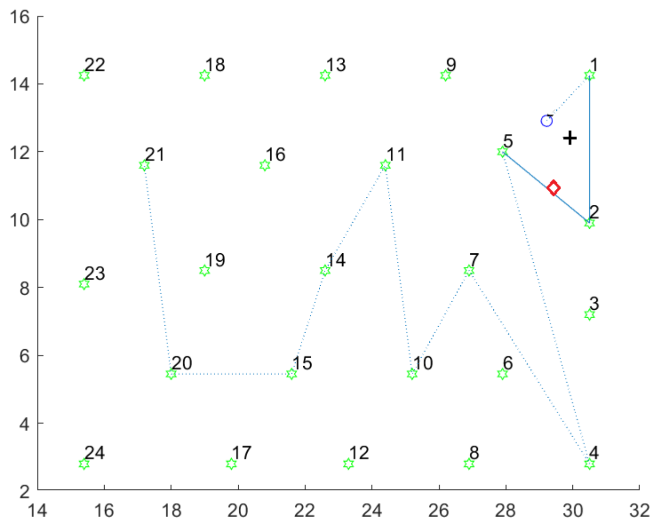


FIGURE 4.5: Effect of farther beacons on position estimate

Figure 4.5 shows an example for the proposed algorithm. In this figure, green stars are beacon positions. Blue circle is true position. Dotted blue line connects the true positions to beacons in descending order of RSS. Red diamond is the position output using WC method using all detected beacons (k = n). Solid blue line represents a well-conditioned triangle. The black plus sign represents the position computed using proposed method.

Position estimate using 3 closest beacon is better than estimates using farther beacons.

Algorithm 1: Selection of beacons using well-conditioned triangle

```
Input: BLEBeacons = table of beacon's ID, location(X, Y), major and minor values
       BLEscan = BLEfingerprint
Output: Well-conditioned triangle
Start;
sort BLEscan in descending order of RSS;
if number of detected beacons < 3 then
    output null;
    break;
else
    selection = select first 3 beacons from BLEscan;
    /* Set list of well-conditioned triangles to empty */
    WCTlist = [];
    while WCTlist is empty do
        DT = DelaunayTriangulation(selection.location);
        // generate triangulation network from selected beacons
           location
        for triangle in DT do
            if triangle satisfy well-condition then
                add triangle to WCTlist;
                break;
            end
        end
        if number of triangles in WCTlist = 1 then
            compute position estimate;
        else
            if number of triangles in WCTlist > 1 then
                compute position estimate for each triangle;
                average position estimate;
            else
                if all beacons used then
                    output null;
                    break;
                else
                    add next beacon to selection
                end
            end
        end
    end
end
```

Stride-Length and Heading for IMU

This method is based on the algorithm proposed by Weinberg [46]. When people move, there is vertical movement of body in each step. Weinberg [46] used vertical acceleration to detect step events and Stride-length for each step was computed by an empirical formula given in equation 4.3. Gyroscope was used to estimate heading at each step.

$$SL = 2K * (max(accmag_{step}) - min(accmag_{step}))^1/4 \tag{4.3}$$

Algorithm 2: Stride-length and Heading Algorithm

```
Input: acc = Accelerometer data and gyr = Gyroscope Data
Output: Stride-Lengths and Headings
Start;
/* Compute Stride-Lengths                                    */
Compute magnitude of acceleration from all the components and store it -> accmag;
Perform low-pass filter on the computed magnitude ;
set lower and upper acceleration threshold;
for each accmag do
    if accmag > lower threshold and accmag < upper threshold then
        mark as motion start
    else
        if accmag < - lower threshold then
            if previous state is in motion then
                mark as motion
            else
                mark as motion stop
            end
        else
            mark as in motion
        end
    end
end
for each motion start do
    Estimate Stride-length using Weinberg expression given by 4.3
end
/* Compute Headings                                          */
Calculate initial row, pitch and yaw values Create device to global rotation matrix for each
 gyr do
    Update rotation matrix with gyr values
end
for each motion start do
    compute heading from rotation matrix
end
Output Computed Stride-Lengths and headings
```

Sensor Fusion

Kálmán filter is a popular method for sensor fusion. System variables for example positions are modelled as state variables. Real observations of those variables are modelled as observation state. Kálmán filter works in two phases i) Predict Phase and ii) Update Phase. Predict phase applies a transition model to push one state to another state. This phase also computes the predicted co-variance in new state. In the update phase, newly computed states are combined with observation state using Kálmán gain to output a filtered estimate. Kálmán gain is a weighting factor calculated on basis of error co-variance of the transition model and the observation model. It tells how much to change the predicted state to reflect an observed state.

Position estimate from stride-length and heading are suitable for prediction phase as a new estimate is calculated from old estimate. Wi-Fi and BLE estimates are independent to previous estimates hence a transition model is not possible. This makes it unfit for prediction phase. On the other hand, Position estimates from Wi-Fi and BLE are suitable for observation state as they provide a stable way to constraint error from prediction phase. The *GetSensorData* app collects IMU data in higher frequency than BLE and Wi-Fi data combined. This means that between two estimates from either of the network-based solutions, there are many position estimates provided by inertial-based solution. Hence, multiple prediction phases occur between two update phases. Kálmán filter allows this but a mechanism to detect which phase to execute is required. A mechanism based on timestamp was devised to trigger correct phase execution.

4.3.3 How to measure the positioning error

Although, Fréchet distance is popular measure to compare similarity between two tracks. It was not used. The reasons are i) Fréchet distance uses minimum distance between the tracks and ii) there is no one point to one point association between the compared tracks. This is not preferable as error in each prediction is required for analysis. Association of prediction position with multiple true positions makes the result ambiguous. Hence, another way for comparison was produced. Error is the euclidean distance between position estimate and true position. Position estimates are obtained from positioning methods described above. True positions are interpolated from the test track. At start of track. GetSensorData app starts recording but there is some wait time before walking start. At the end the app records some extra readings due to the delay in finishing track and pressing stop button. However, exact start time, direction change and exact end time are marked and all the readings have associated timestamp value. Using the marks it is possible to interpolate true position at any intermediate timestamp. A script was made which take list of timestamps as input and output coordinates at those timestamps. Wi-Fi sensor gave exact timestamp for each fingerprint, for BLE timestamps were derived from the window size and update interval values, for IMU timestamps of step detection were used. Now that estimated position and true position were known, error was calculated.

This chapter details on the experiment environment, software used for data collection, procedure followed during collection and format of collected data. Generating fingerprint from RSS data is also shown. For BLE, two methods of using window is discussed along with three techniques for handling redundant BLE advertisements. Proposed method of selecting a well-conditioned triangle for BLE positioning is described in detail. Implementation of weighted centroid for position estimation is explained in brief. Stride-length and heading estimation method for IMU positioning is comprehensively described. Sensor fusion technique using Kálmán filter is stated. Application details of Kálmán filter in current context is described without going in depth on mechanism of Kálmán filter. Finally, method used for calculating error is described highlighting the unsuitability of popular Fréchet distance method.

Chapter 5

Results and Discussion

This chapter deals with the obtained results and discussion on those results. First section presents results obtained from Wi-Fi Fingerprinting. second section discusses the outcomes and effects of the studied factors on BLE positioning. Third section delineates about the results from IMU positioning and finally last section draws up the results of fusion.

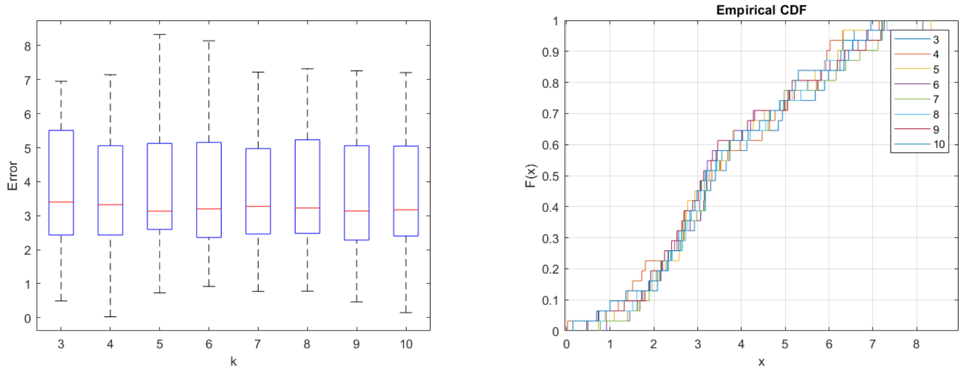
5.1 Wi-Fi position

Thought there are numerous Wi-Fi positioning methods available, k-NN based fingerprinting method was used for this work. k-NN comprises **k** as value and **nn** as nearest neighbour meaning the fingerprints value of the nearest neighbour used for position estimation. In this work, positioning results were tested for k values ranging 3 through 10. Figure 5.2 shows results of Wi-Fi Positioning (for k = 4 and 7). The Lowest average error was observed at k-value 4 and highest average error was observed at k-value 7. Table 5.1 presents an overview of errors obtained at different k-values. No significant changes in

K Value	Average Error	Standard Deviation	75 percentile	Count	Standard Error
3	3.806	1.806	5.506	31	0.324
4	3.608	1.841	5.059	31	0.330
5	3.709	1.813	5.125	31	0.325
6	3.746	1.824	5.154	31	0.327
7	3.834	1.853	4.973	31	0.332
8	3.732	1.825	5.233	31	0.327
9	3.629	1.874	5.059	31	0.336
10	3.627	1.866	5.049	31	0.335

TABLE 5.1: Overview of errors at different k values for k-NN fingerprinting

errors at different k-values was found. The average error was above 3.5 m in all cases. CDF plot 5.1b of errors shows that k-value did not significantly affect error in individual estimates. In figure 5.2, the green lines show the test track and the blue dots represent



(A) Error distribution by k value

(B) CDF of Error at different k value

FIGURE 5.1: Distribution of error in Wi-Fi estimates

the estimated positions. The red line represent the estimated track. It joins the position estimates in ascending order of time. It can be inferred from figure 5.2 that Wi-Fi estimates were not able to satisfactorily represent the test track. The number of positions estimated (count), average of the errors in each estimation, 75th percentile of the errors from each estimation (P75 error), standard deviation(std.) and standard error of the mean (SE) are available in the figure. Whereas the standard deviation provides the degree to which individuals recordings deviate from the mean value. SE is an estimate of deviation of sample mean from the population mean. SE tends to zero with increase in sample size, as large sample size improves the estimate of the population mean, It is computed by the formula 5.1.

$$SE_x = \frac{s}{\sqrt{n}} \tag{5.1}$$

where:

s = Standard Deviation
n = Number of estimates

5.2 BLE position

Unlike Wi-Fi sensor, BLE sensor can provide output after each advertisement is detected. This is beneficial for accuracy enhancement but comes with own issues. Positioning is not feasible with a single advertisement. So, BLE advertisements need to be evaluated in

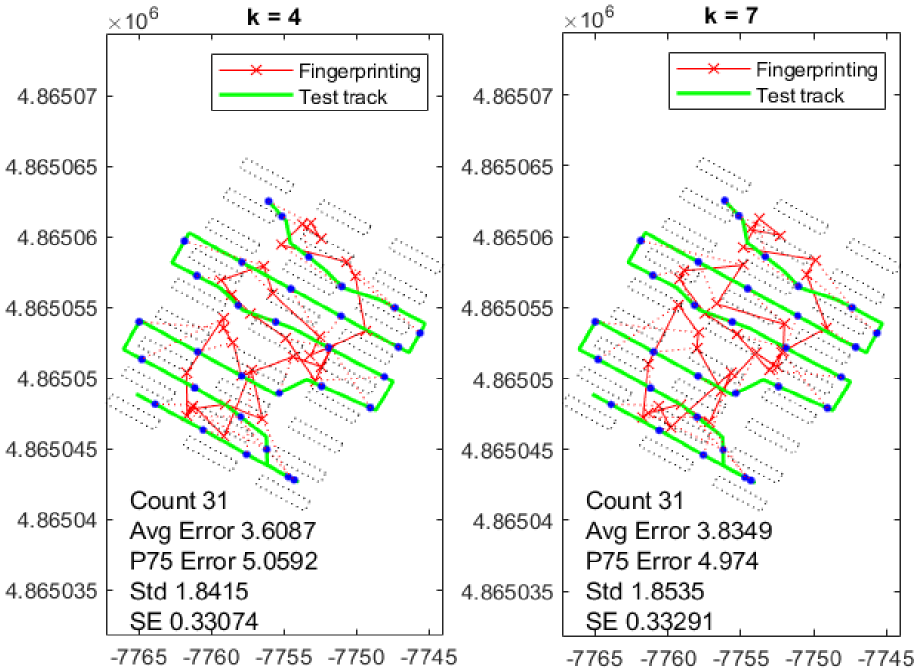
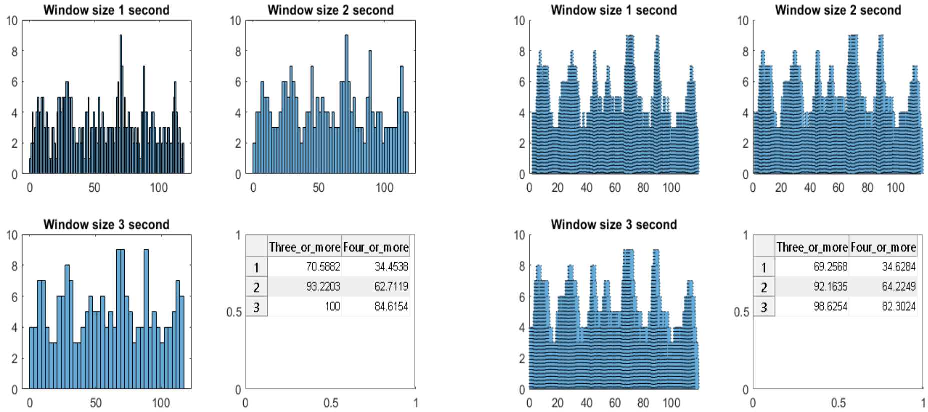


FIGURE 5.2: Position estimates from k-NN fingerprinting

a group. Two techniques for grouping advertisements have been tested: (a) Discrete Time Window and (b) Continuous Time Window. The methods are described in Methods and data-sets chapter. Once advertisements are grouped, another issue arises. A window may contain more than one advertisement from same beacon. Three strategies for handling such redundant advertisements are tested: selecting either the i) last, ii) average or iii) highest of redundant advertisements. Result of BLE position estimate was studied on four factors.

1. Choice of window size (1,2 or 3 seconds)
2. Choice of window technique (Discrete or Continuous)
3. Choice of redundant advertisement handling strategy (Last, Average or Highest)
4. Choice of positioning algorithm (Weighted Centroid or Proposed method)

Window size was a significant factor affecting positioning. In smaller window sizes, enough advertisements may not be collected. Large size smears the position estimate [15]. To understand the advertisements collection over time, unique advertisements were plotted against time. Unique advertisements are plotted instead of all advertisements because, the redundant advertisements were combined using one of the redundant advertisement handling strategies. Position estimation requires at least 3 unique advertisements (an exception to this is proximity estimation which can work with one advertisement). Sizes should be selected where at least 3 unique advertisements are available in the majority of the windows.



(A) Discrete windows (B) Continuous windows

FIGURE 5.3: Unique advertisement recorded in different window sizes. Lower right plot shows percentage of windows that detected 3 or more and 4 or more unique advertisements respectively

Figure 5.3a shows the number of unique advertisements detected in each discrete window. A discrete window start after previous window expires, and they do not overlap each other. Size less than 1 second had single advertisements in the majority of windows. For 1-second window size, about 70% of windows had more than 3 unique advertisements. The percentage rose to 93% for 2-second window size and 100% for 3-second window size. Similarly, percentage of 4 or more unique advertisements were 34%, 63% and 84% in window sizes 1,2 and 3 seconds respectively. Continuous windows start at regular interval. The interval is kept smaller than the window size, due to this continuous windows overlap each other. Interval for continuous window was set at 0.5 second in all window sizes. Intervals smaller than 0.5 seconds took high computation time but result were similar. Result of advertisement detection using continuous window grouping technique showed similar percentage of detection as discrete window technique.

Figure 5.4 presents the positioning estimates with discrete window and weighted centroid method (k-value 3) using last of redundant advertisements. From the figure it was evident that the predicted path was more similar to the original path than the path predicted by Wi-Fi. Average error in BLE estimates is lower than that of Wi-Fi. Number of estimates from BLE were also higher than from Wi-Fi. Table 5.2 display overview of error with BLE position on different combinations of the four factors listed before. The proposed algorithm used 3 beacons for position estimation so k-value for WC method was also set as 3 in all WC estimations.

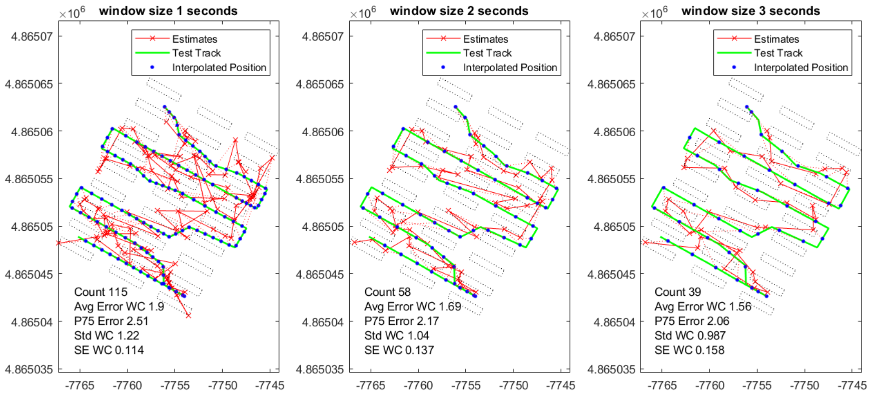


FIGURE 5.4: Example of position estimate with discrete window and WC method

Figure 5.5 shows error distribution of the studied factors. On analysing the effect of each factor, it was observed that the choice of window technique had no drastic effect on accuracy. The average difference of average error was found to be 0.005m. Among the strategies for advertisement repeats, highest of the BLE repeats had the least error. It was followed by average of repeats. Last of the repeats had higher average error. Except on window size 3 using discrete window, error from highest of BLE was lower than other techniques. Proposed positioning method had similar result but lower variance than WC method. One-second window size fared higher errors than it's counterparts. Window sizes of 2 and 3 seconds had improved results than 1 second. Window size influences effect of other factors on the error. It is reasonable to study affect of the factors independent to window size. Figure 5.6 visualizes the error distribution of the factors segregated by window size. Effect of grouping advertisements in continuous window or discrete window varied by window size. Discrete technique show lower variance in larger window sizes. Continuous window technique create double the number of windows than discrete method (because update interval is set at 0.5). This could be the cause of large variance in continuous window technique as it has more samples to work with. The average error were similar in both techniques. Among the different repeat handling strategy, highest of repeats was better than other strategies. The effect of different strategies was more profound in window size 2 seconds. Average of repeats and Last of repeats decrease

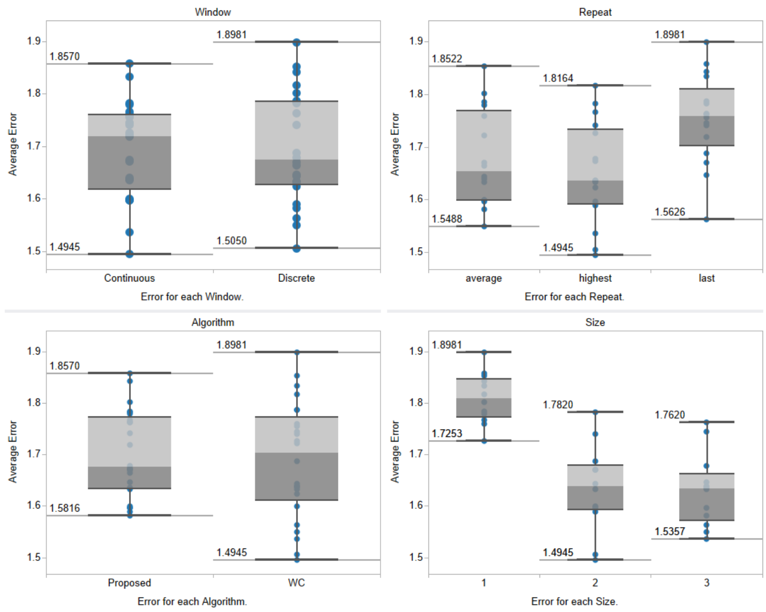


FIGURE 5.5: Error distribution for each factor

error with increasing window size but highest of BLE shows anomaly. Error is least in window size 2 seconds. Increase in error from 2 second to 3 second suggests smearing of position space meaning the device had moved considerably in 3 seconds. Hence, it can be inferred that the use of the highest repeats is not suitable in window sizes larger than 2 second. The proposed algorithm had lower variance in error distribution than WC method although the average error was larger. On comparison of the algorithms for one second and two seconds window. It seems that the proposed algorithm was able to reduce larger error. However, it seemed to be missing out on those windows where WC algorithm was getting smaller error. In three-second windows, the distribution was similar for both proposes and WC algorithms but errors for WC algorithm was lower. Proposed algorithm rejects windows when only two beacons are detected. WC algorithm is capable of estimating position with two beacons but the proposed algorithm requires at least three. Further investigation revealed that percentage of windows with exactly two unique beacons detected in discrete technique was 20%, 6.7% and 0% respectively in one second, two seconds and three seconds window. In continuous window technique it was 19.4%, 6.3% and 1.2% respectively i.e. the percentages were similar in both techniques. If WC algorithm was getting smaller error due to positioning with two beacons, error in one-second window should have been smaller than in two-second windows because percentage of exactly two beacons detected is higher in one-second window. Hence, it can be concluded that WC algorithms ability to estimate position with only two beacons did

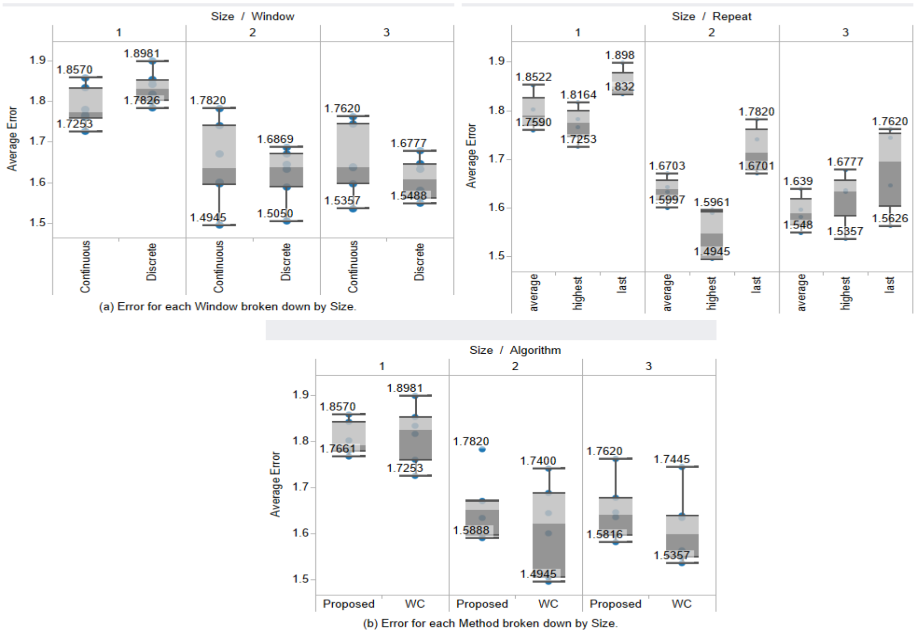


FIGURE 5.6: Error distribution due to each factor. Distribution is separated by window size.

not contribute to smaller error. The proposed algorithm rejects beacon with higher RSS value if it does not form a well-conditioned triangle with other beacons. WC algorithm does not reject beacon with higher RSS. This seems to have influenced the output.

5.3 IMU position

IMU positions estimate was close to the test track. The results were highly sensitive to sensor bias. Small error in bias determination would produce different result. Estimates were found to be sensitive to the rate of change of angle. Taking fast turns disorients the heading estimations. IMU estimates were very accurate over short distances. The figure 5.8 shows the estimates are less spread than in other methods. Distance estimation was not very accurate due to which average error is high. Heading estimations were very accurate. It is important to note that the device orientation is based on magnetic north. For real world applications conversion to true north is essential otherwise estimates will be misaligned.

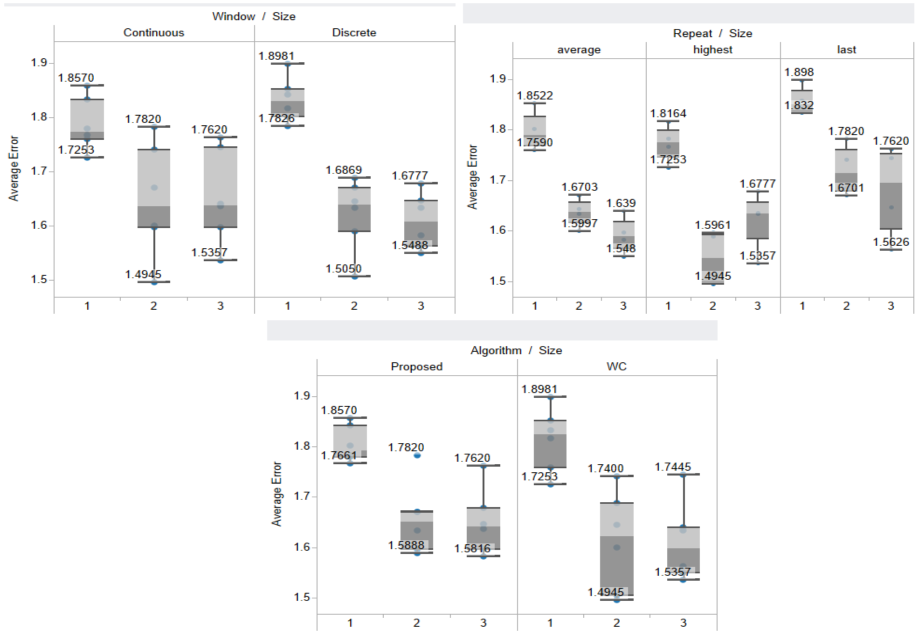


FIGURE 5.7: Error distribution due to window size on each factor.

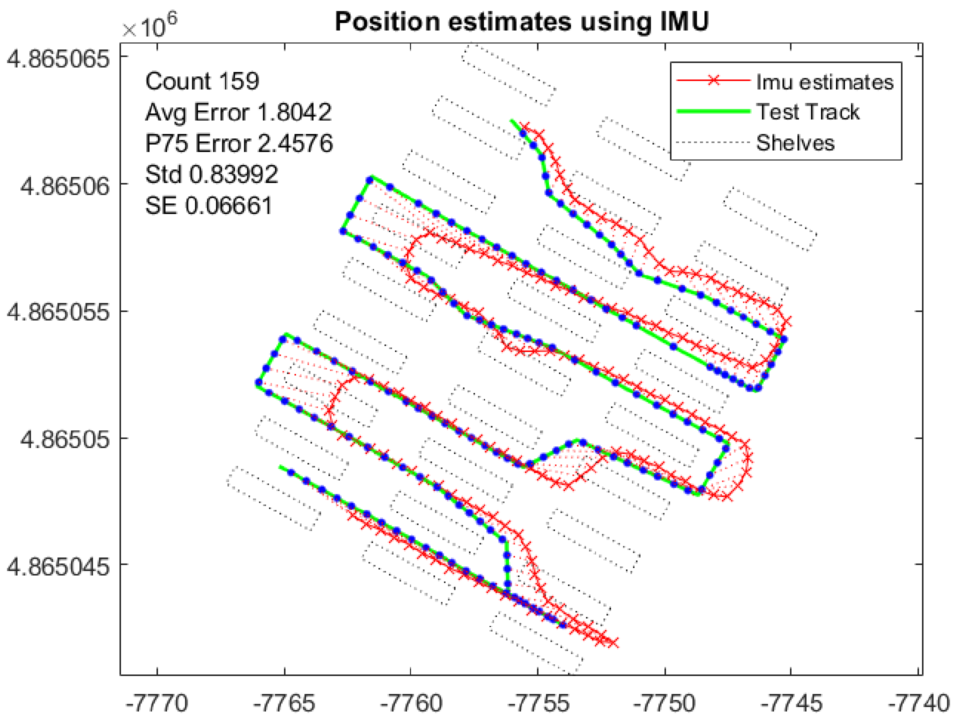


FIGURE 5.8: Position estimates with discrete Stride-length and heading method

5.4 Sensor Fusion

IMU Position estimates were fused with estimates from BLE on window size 2 second, continuous window grouping, highest of redundant advertisement and WC algorithm. Output of fusion had lower average error than either of IMU or BLE. Fusion results were able to accurately represent the turns.

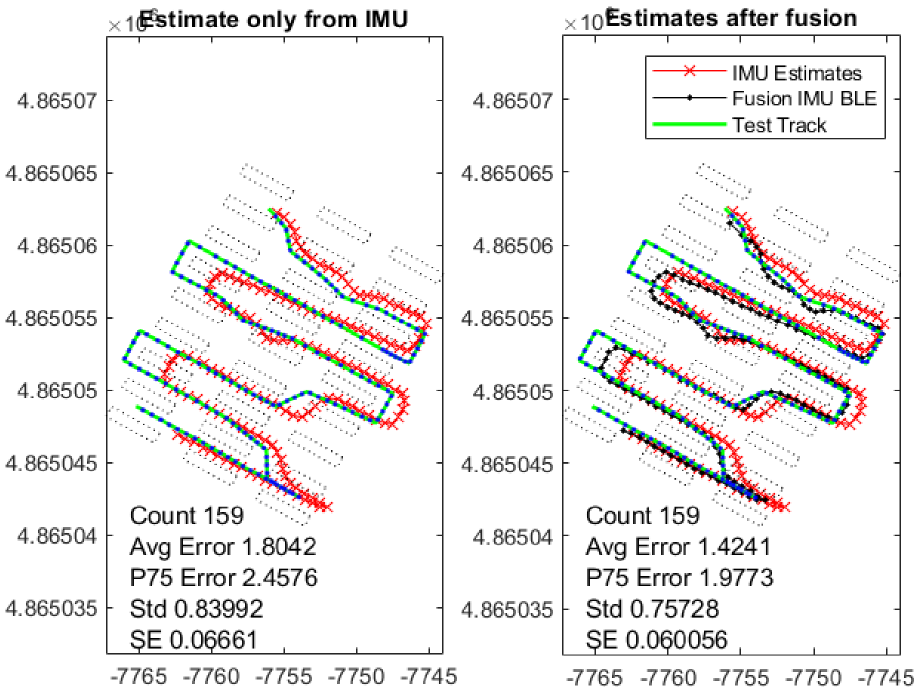


FIGURE 5.9: Position estimates with fusion of IMU and BLE

This chapter presented output from IPS systems used in this research. Accuracy of Wi-Fi positioning was found similar to error reported by other Wi-Fi based IPS literature. BLE based systems performed better than Wi-Fi. Choice of discrete or continuous window had minor effect. Use of highest of redundant advertisements had highest accuracy but the positions appeared to be smeared in window size 3 seconds.it is recommended to further study the relation with window size. New proposed method did not show significant improvement in accuracy but variance in error is less than WC method. Further improvements seem necessary.

Window Size	Window Technique	Redundant	Algorithm	Average Error	P75 Error
1	Discrete	average	WC	1.852	2.406
			Proposed	2.045	2.406
		highest	WC	1.816	2.249
			Proposed	2.106	2.249
		last	WC	1.898	2.483
			Proposed	2.098	2.483
	Continuous	average	WC	1.758	2.406
			Proposed	1.778	2.406
		highest	WC	1.725	2.249
			Proposed	1.766	2.249
		last	WC	1.832	2.483
			Proposed	1.857	2.483
2	Discrete	average	WC	1.643	2.309
			Proposed	1.944	2.309
		highest	WC	1.504	2.079
			Proposed	2.028	2.079
		last	WC	1.686	2.352
			Proposed	2.073	2.352
	Continuous	average	WC	1.599	2.309
			Proposed	1.670	2.309
		highest	WC	1.494	2.079
			Proposed	1.596	2.079
		last	WC	1.739	2.352
			Proposed	1.781	2.352
3	Discrete	average	WC	1.548	2.198
			Proposed	2.283	2.198
		highest	WC	1.632	2.160
			Proposed	2.207	2.160
		last	WC	1.562	2.344
			Proposed	2.386	2.344
	Continuous	average	WC	1.639	2.198
			Proposed	1.596	2.198
		highest	WC	1.535	2.160
			Proposed	1.635	2.160
		last	WC	1.744	2.344
			Proposed	1.761	2.344

TABLE 5.2: Overview of errors on all combination of factors.

Chapter 6

Conclusion and Future Works

6.1 Conclusion

The main objective of the research work in this thesis is to improve the accuracy positional accuracy of BLE based positioning system. Network based and inertial based IPS systems were reviewed. An existing BLE positioning environment was selected for experiment. Wi-Fi fingerprinting database and BLE fingerprint dataset were publicly available. The datasets provide necessary tools to reproduce the results, so they are good for learning.

Thesis objective presented in chapter 1 is discussed here, with its explanations.

1. To study and analyse factors affecting BLE positioning

   Four factors affecting BLE positioning were studied (window size, window technique, redundant advertisements handling strategies and algorithms). All combinations of 3 different window sizes, 2 window techniques, 3 redundant advertisements handling strategies and 2 algorithms were tested. 36 different combinations tested. A new algorithm was also proposed. Window size mattered to some extent. 2 seconds window had improvement over 1-second window size but results from 3 seconds window size was similar to 2 seconds. So, the choice is between 1 second or more than 1 second. Results show that choice of window technique had no significant effect. Update interval's effect on accuracy was not found but it affects the amount of computation. Update interval should depend on available computing resource. Discrete window can be taken as a special case of continuous window so, a good approach is using continuous window with a suitable update interval. Choice of this would depend on specific use case. Handling strategies for redundant advertisement in same window is another studied factor. Using highest of BLE showed best results but it might be misleading in larger window size. Smearing was seen in window size 3. Other strategies (Average of repeats and Last of repeats) did not detect smearing.

2. To design and implement a positioning algorithm based on well-conditioned triangle for BLE positioning and study its effects.

   Proposed algorithm of selecting beacons based on well-conditioned triangle had no improvements on accuracy but had improved consistency than WC algorithm. Further study is required to understand its working.

3. To implement an integration of BLE positioning and inertial method.

   Literature suggested sensor fusion approach provided improved accuracy or functionality so it was chosen for implementation. IMU was chosen to be fused with BLE and IMU sensors are available in most smartphones. This would be infrastructure-free and no extra setups are required. Kálmán filter offered a simple, intuitive but powerful mechanism for fusion. It was adapted to suit the research scenario. An experiment was designed for comparing accuracy before and after fusion. Existing system was used as baseline. Sensor fusion results were more accurate than BLE. Additionally, the fusion results were better at turns where other IPS showed more error.

Overall, for BLE positioning window size of two seconds seems to be best. Larger window size are prone to smearing. Continuous window technique provides a general. Update interval should be balanced on available computation power and update frequency. Strategy of using highest of redundant advertisements performed the best. If smearing can be reduced, it is a viable option. IMU provides high accuracy estimates. With the fusion of BLE and IMU, IMU can provide estimates during BLE positioning intervals and BLE estimates can correct drift error. The overall fusion system performed well.

6.2 Future Works

Results presented are not conclusive and leaves much scope for improvement. Some ways to improve the result are suggested.

- Although beacon selection is not based on RSS value, position estimation still used RSS for computing weights. RSS values have error, other techniques for computing position should be explored.

- Device calibration stage was not explicit. Calibration can improve IMU accuracy.

- Proposed approach depends on detection of the closest beacons. In some cases the closest beacon was not detected,
- Count of number of redundant advertisement also gives clue to closeness.
- Well-conditioned triangle is simple technique. Braced quadrilaterals and centred polygons are more robust than triangles. These can be explored.

SENSOR FUSION OF IMU AND BLE USING A WELL-CONDITION TRIANGLE APPROACH FOR BLE POSITIONING

Amrit KARMACHARYA

21 february

Appendix A

BLE positions

A.1 BLE positions using Discrete time window and Weighted centroid algorithm

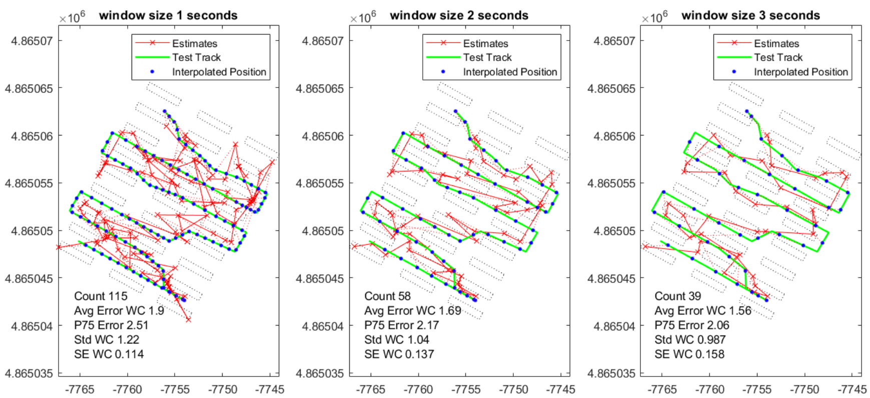


FIGURE A.1: Last of redundant advertisements

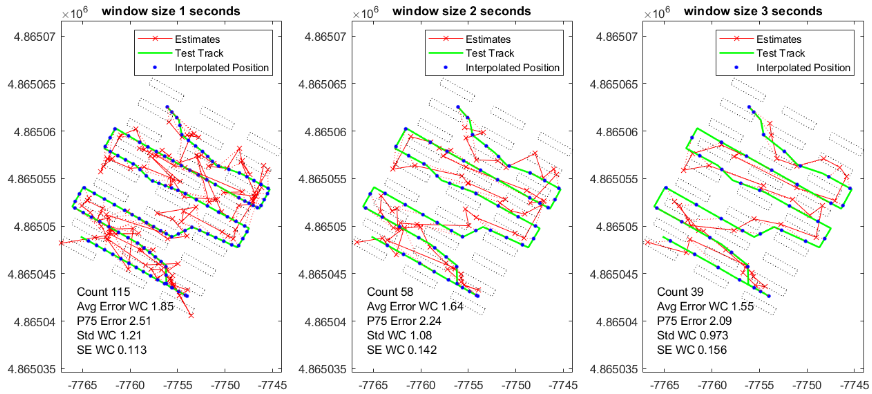


FIGURE A.2: Average of redundant advertisements

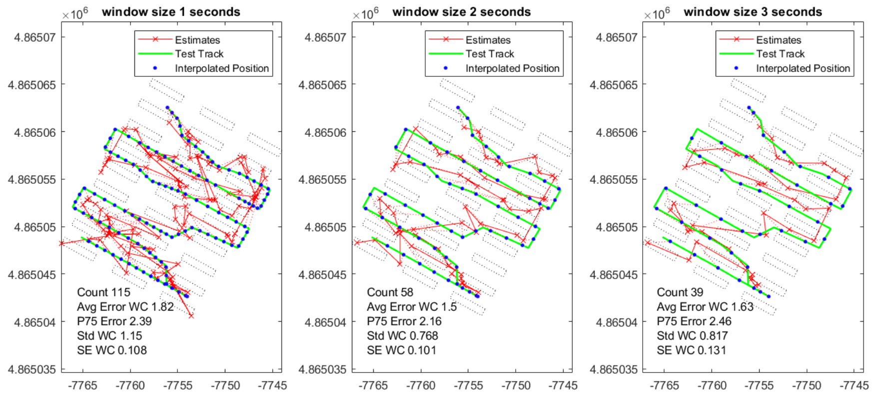


FIGURE A.3: Highest of redundant advertisements

A.2 BLE positions using Continuous time window and Weighted centroid algorithm

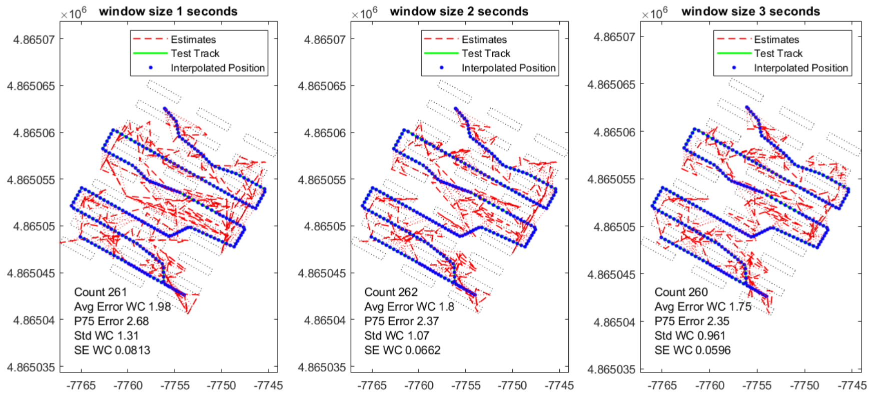


FIGURE A.4: Last of redundant advertisements

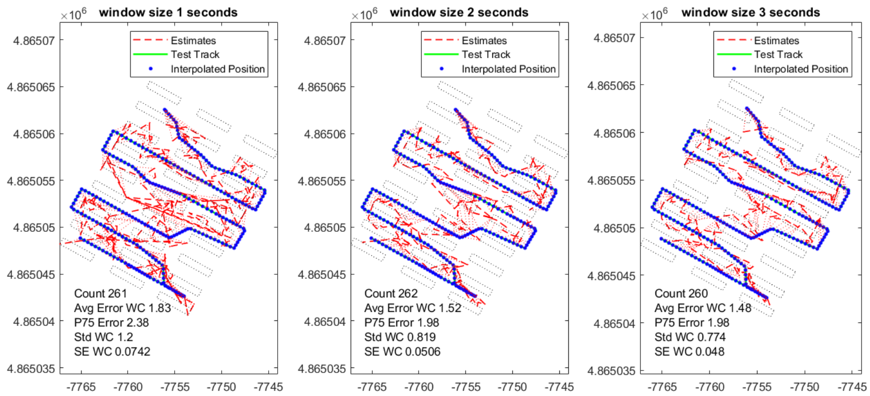


FIGURE A.5: Average of redundant advertisements

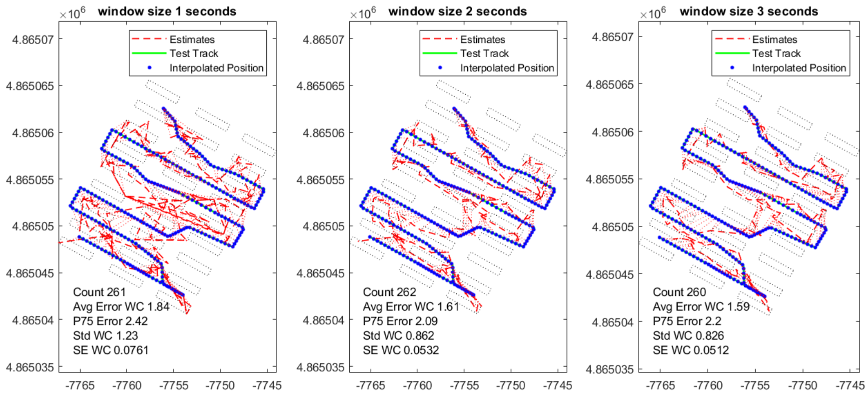


FIGURE A.6: Highest of redundant advertisements

A.3 BLE positions using Discrete time window and Proposed algorithm

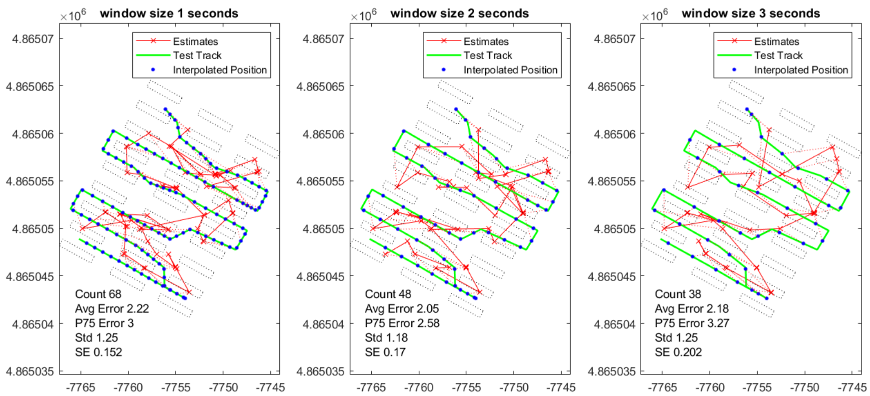


FIGURE A.7: Last of redundant advertisements

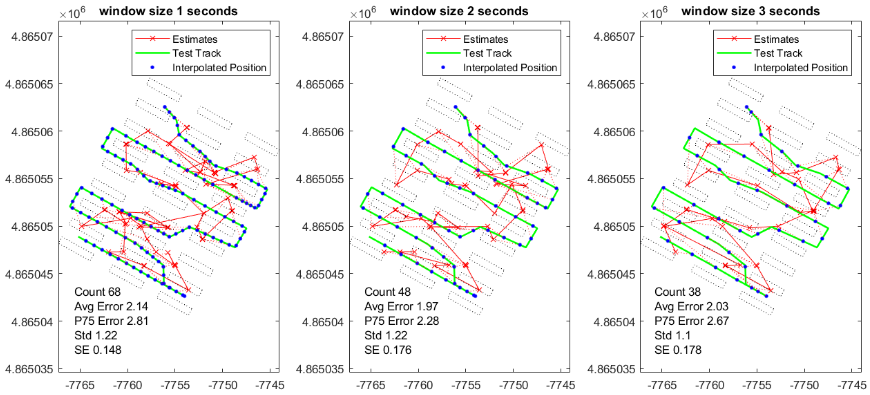


FIGURE A.8: Average of redundant advertisements

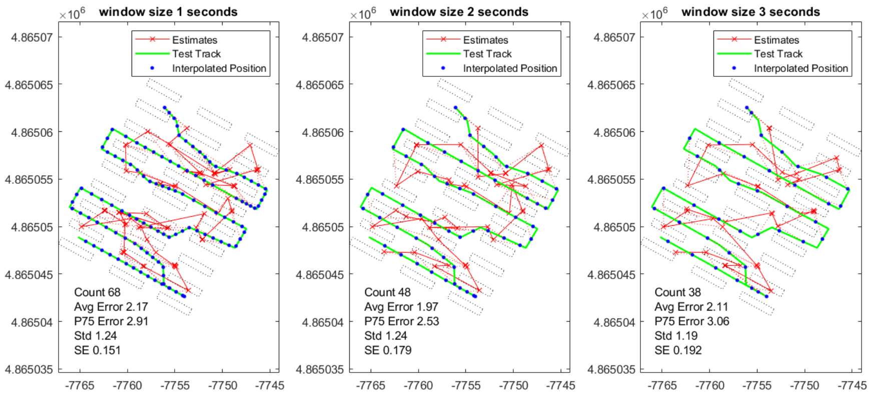


FIGURE A.9: Highest of redundant advertisements

A.4 BLE positions using Continuous time window and Proposed algorithm

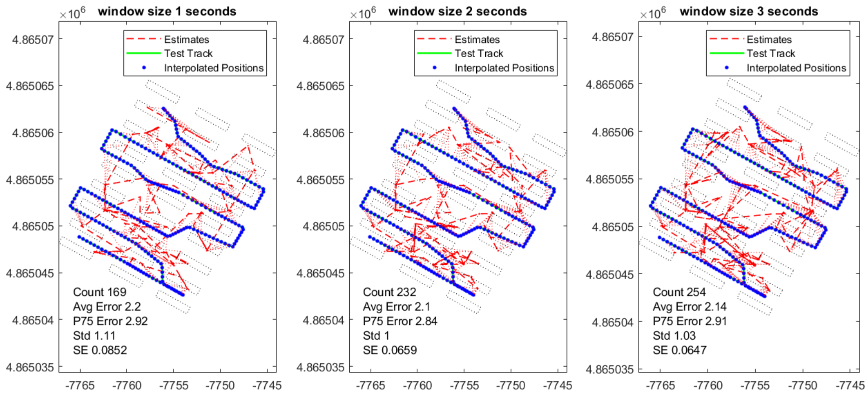


FIGURE A.10: Last of redundant advertisements

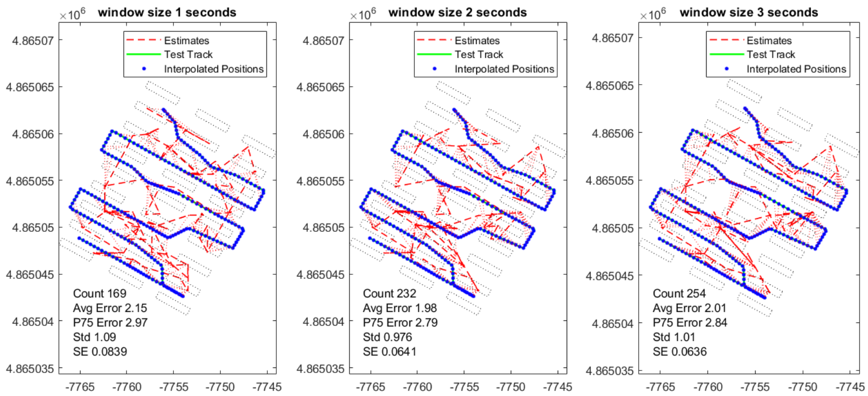


FIGURE A.11: Average of redundant advertisements

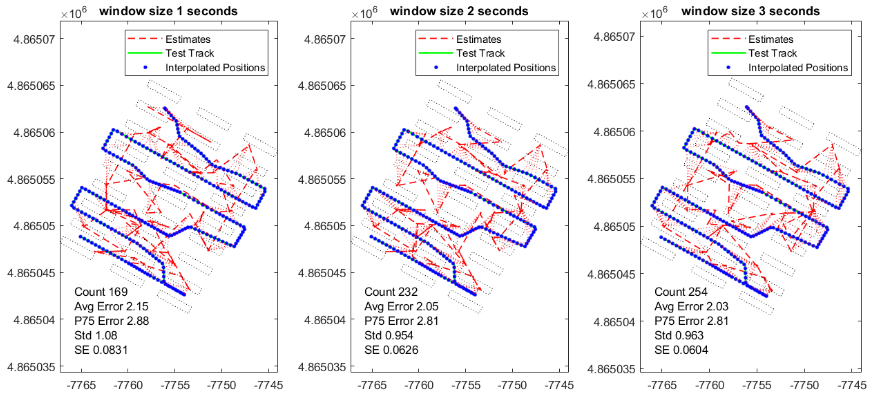


FIGURE A.12: Highest of redundant advertisements

Bibliography

[1] Mohammad OA Aqel et al. "Review of visual odometry: types, approaches, challenges, and applications". In: *SpringerPlus* 5.1 (2016), p. 1897.

[2] Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury. "SurroundSense: Mobile Phone Localization via Ambience Fingerprinting". In: *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*. MobiCom '09. New York, NY, USA: ACM, 2009, pp. 261–272. ISBN: 978-1-60558-702-8. DOI: 10.1145/1614320.1614350. URL: http://doi.acm.org/10.1145/1614320.1614350.

[3] Anahid Basiri et al. "Indoor location based services challenges, requirements and usability of current solutions". In: *Computer Science Review* 24 (2017), pp. 1 – 12. ISSN: 1574-0137. DOI: https://doi.org/10.1016/j.cosrev.2017.03.002. URL: http://www.sciencedirect.com/science/article/pii/S1574013716301782.

[4] Ashok Kumar Jain B.C.Punmia and Arun Kumar Jain. *Surveying Volume 2*. Vol. II. Laxmi Publications, 1994, pp. 356–359.

[5] SIG Bluetooth. "Bluetooth core specification version 4.0". In: *Specification of the Bluetooth System* 1 (2010), p. 7.

[6] Vicente Cantón Paterna et al. "A bluetooth low energy indoor positioning system with channel diversity, weighted trilateration and kalman filtering". In: *Sensors* 17.12 (2017), p. 2927.

[7] Zhenghua Chen et al. "Fusion of WiFi, smartphone sensors and landmarks using the Kalman filter for indoor localization". In: *Sensors* 15.1 (2015), pp. 715–732.

[8] Krishna Chintalapudi, Anand Padmanabha Iyer, and Venkata N Padmanabhan. "Indoor localization without the pain". In: *Proceedings of the sixteenth annual international conference on Mobile computing and networking*. 2010, pp. 173–184.

[9] P Connolly and D Boone. "Indoor location in retail: Where is the money". In: *Business Models Analysis Report* (2013).

[10] Alejandro Correa et al. "A review of pedestrian indoor positioning systems for mass market applications". In: *Sensors* 17.8 (2017), p. 1927.

[11] Pavel Davidson and Robert Piché. "A survey of selected indoor positioning methods for smartphones". In: *IEEE Communications Surveys & Tutorials* 19.2 (2016), pp. 1347–1370.

[12] Meenakshi D Dhobale et al. "A Review on Indoor Navigation Systems and Technologies for Guiding the Users". In: *International Journal of Scientific Research in Computer Science, Engineering and Information Technology* (2019).

[13] Estefania Munoz Diaz, Dina Bousdar Ahmed, and Susanna Kaiser. "A Review of Indoor Localization Methods Based on Inertial Sensors". In: *Geographical and Fingerprinting Data to Create Systems for Indoor Positioning and Indoor/Outdoor Navigation*. Elsevier, 2019, pp. 311–333.

[14] Lee J Easson. "Indoor Localization Using a Smartphone: Approaches, Issues, and Challenges". PhD thesis. University of Mississippi, 2019.

[15] Ramsey Faragher and Robert Harle. "Location fingerprinting with bluetooth low energy beacons". In: *IEEE journal on Selected Areas in Communications* 33.11 (2015), pp. 2418–2428. URL: http://smedia.ust.hk/james/projects/people_aware_smart_city_applications/paper/1.pdf.

[16] V Fox et al. "Bayesian filtering for location estimation". In: *IEEE pervasive computing* 2.3 (2003), pp. 24–33.

[17] Dongsoo Han, Suk-hoon Jung, and Sangjae Lee. "A sensor fusion method for Wi-Fi-based indoor positioning". In: *ICT Express* 2.2 (2016), pp. 71–74.

[18] Pavel Ivanov, Matti Raitoharju, and Robert Piché. "Kalman-type filters and smoothers for pedestrian dead reckoning". In: *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE. 2018, pp. 206–212.

[19] Antonio R Jiménez, Fernando Seco, and Joaquín Torres-Sospedra. "Tools for smartphone multi-sensor data registration and GT mapping for positioning applications". In: *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE. 2019, pp. 1–8.

[20] Hee-Yong Kang et al. "Study of a Hybrid Algorithm for Indoor Positioning". In: *International Journal of Control and Automation* 11.8 (2018), pp. 25–38.

[21] Jonghyuk Kim and Salah Sukkarieh. "Real-time implementation of airborne inertial-SLAM". In: *Robotics and Autonomous Systems* 55.1 (2007), pp. 62–71. URL: http://users.cecs.anu.edu.au/~Jonghyuk.Kim/pdf/RAS-AirborneSLAM_v2.3.pdf.

[22] Christopher Langlois, Saideep Tiku, and Sudeep Pasricha. "Indoor localization with smartphones: Harnessing the sensor suite in your pocket". In: *IEEE Consumer Electronics Magazine* 6.4 (2017), pp. 70–80.

[23] Jae-Gwang Lee, Seoung-Hyeon Lee, and Jae-Kwang Lee. "Design of the Model for Indoor Location Prediction Using IMU of Smartphone Based on Beacon". In: *International Conference on Software Engineering Research, Management and Applications*. Springer. 2018, pp. 161–173.

[24] Alexander Lindemann et al. "Indoor Positioning: A Comparison of WiFi and Bluetooth Low Energy for Region Monitoring." In: *HEALTHINF*. 2016, pp. 314–321.

[25] Elena Simona Lohan et al. "Received signal strength models for WLAN and BLE-based indoor positioning in multi-floor buildings". In: *2015 International conference on location and GNSS (ICL-GNSS)*. IEEE. 2015, pp. 1–6.

[26] Guoqiang Mao, Barış Fidan, and Brian DO Anderson. "Wireless sensor network localization techniques". In: *Computer networks* 51.10 (2007), pp. 2529–2553.

[27] MathWorks. *Delaunay triangulation*. URL: https://www.mathworks.com/help/matlab/math/delaunay-triangulation.html.

[28] Fazeelat Mazhar, Muhammad Gufran Khan, and Benny Sällberg. "Precise indoor positioning using UWB: a review of methods, algorithms and implementations". In: *Wireless Personal Communications* 97.3 (2017), pp. 4467–4491. URL: https://idp.springer.com/authorize/casa?redirect_uri=https://link.springer.com/article/10.1007/s11277-017-4734-x&casa_token=MNBT7qVlWiUAAAAA:u1bUBhqDgRYSmzZ9QmTVeUMM_ggFdh-awplOFKbzVNiz-U67PeT73HnLxeob2_1yGOQGIQC2EwEBBNcDYA.

[29] Germán Mendoza-Silva et al. "Long-term WiFi fingerprinting dataset for research on robust indoor positioning". In: *Data* 3.1 (2018), p. 3.

[30] Germán Martín Mendoza-Silva, Joaquín Torres-Sospedra, and Joaquín Huerta. "A Meta-Review of Indoor Positioning Systems". In: *Sensors* 19.20 (2019), p. 4507.

[31] Germán Martín Mendoza-Silva et al. "BLE RSS Measurements Dataset for Research on Accurate Indoor Positioning". In: *Data* 4.1 (2019), p. 12.

[32] Mozilla. *Mozilla Location Service*. Mozilla. Jan. 2020. URL: https://location.services.mozilla.com/.

[33] Mario Muñoz-Organero, Pedro J Muñoz-Merino, and Carlos Delgado Kloos. "Using bluetooth to implement a pervasive indoor positioning system with minimal requirements at the application level". In: *Mobile Information Systems* 8.1 (2012), pp. 73–82.

[34] Filippo Palumbo et al. "A stigmergic approach to indoor localization using bluetooth low energy beacons". In: *2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE. 2015, pp. 1–6.

[35] S. Pasricha et al. “LearnLoc: A framework for smart indoor localization with embedded mobile devices”. In: *2015 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*. 2015, pp. 37–44. DOI: 10.1109/CODESISSS.2015.7331366.

[36] Nissanka Bodhi Priyantha. “The cricket indoor location system”. PhD thesis. Massachusetts Institute of Technology, 2005. URL: https://dspace.mit.edu/bitstream/handle/1721.1/33924/67296044-MIT.pdf?sequence=2&isAllowed=y.

[37] Valerie Renaudin et al. “Evaluating Indoor Positioning Systems in a Shopping Mall: The Lessons Learned From the IPIN 2018 Competition”. In: *IEEE Access* 7 (2019), pp. 148594–148628.

[38] Skyhook. *Precision Location*. Skyhook. Jan. 2020. URL: https://www.skyhook.com/precision-location/.

[39] Timo Smieszek, Gianrocco Lazzari, and Marcel Salathé. “Assessing the Dynamics and Control of Droplet-and Aerosol-Transmitted Influenza Using an Indoor Positioning System”. In: *Scientific reports* 9.1 (2019), p. 2185.

[40] Joaquín Torres-Sospedra et al. “Characterising the Alteration in the AP Distribution with the RSS Distance and the Position Estimates”. In: *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE. 2018, pp. 1–8.

[41] Joaquín Torres-Sospedra et al. “Enhancing integrated indoor/outdoor mobility in a smart campus”. In: *International Journal of Geographical Information Science* 29.11 (2015), pp. 1955–1968.

[42] Joaquín Torres-Sospedra et al. “In-home monitoring system based on WiFi fingerprints for ambient assisted living”. In: *Journal of Ambient Intelligence and Smart Environments* 9.5 (2017), pp. 543–560.

[43] Melanija Vezočnik and Matjaz B Juric. “Average step length estimation models’ evaluation using inertial sensors: a review”. In: *IEEE Sensors Journal* 19.2 (2018), pp. 396–403.

[44] B. Wang et al. “Indoor smartphone localization via fingerprint crowdsourcing: challenges and approaches”. In: *IEEE Wireless Communications* 23.3 (2016), pp. 82–89. ISSN: 1558-0687. DOI: 10.1109/MWC.2016.7498078.

[45] Fafa Wang et al. “Indoor Tracking by RFID Fusion with IMU Data”. In: *Asian Journal of Control* 21.0 (2019), pp. 1–10. DOI: 10.1002/asjc.1954. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/asjc.1954. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/asjc.1954.

[46] Harvey Weinberg. “Using the ADXL202 in pedometer and personal navigation applications”. In: *Analog Devices AN-602 application note* 2.2 (2002), pp. 1–6.

[47] Oliver J Woodman. *An introduction to inertial navigation*. Tech. rep. University of Cambridge, Computer Laboratory, 2007.

[48] Feng Ye et al. “A Low-Cost Single-Anchor Solution for Indoor Positioning Using BLE and Inertial Sensor Data”. In: *IEEE Access* 7 (2019), pp. 162439–162453.

[49] Paul K. Yoon et al. “Adaptive Kalman filter for indoor localization using Bluetooth Low Energy and inertial measurement unit”. In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (2015), pp. 825–828.

[50] Shaghayegh Zihajehzadeh et al. “A cascaded Kalman filter-based GPS/MEMS-IMU integration for sports applications”. In: *Measurement* 73 (2015), pp. 200 –210. ISSN: 0263-2241. DOI: https://doi.org/10.1016/j.measurement.2015.05.023. URL: http://www.sciencedirect.com/science/article/pii/S0263224115002821.

Masters
Program
in **Geospatial**
Technologies

SENSOR FUSION OF IMU AND BLE USING A WELL-CONDITION TRIANGLE APPROACH FOR BLE POSITIONING

Amrit Karmacharya

Dissertation submitted in partial fulfilment of the requirements
for the Degree of *Master of Science in Geospatial Technologies*

NOVA IMS Information Management School

Universitat Jaume I

Westfälische Wilhelms-Universität Münster

2020

SENSOR FUSION OF IMU AND BLE USING A WELL-CONDITIONED TRIANGLE APPROACH FOR BLE POSITIONING

Amrit Karmacharya

Masters
Program
in **Geospatial**
Technologies

Supported by:
Education and Culture
ERASMUS MUNDUS